

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2001-177537

(P2001-177537A)

(43)公開日 平成13年6月29日 (2001.6.29)

(51)Int.Cl. ⁷	識別記号	F I	テーマコード(参考)
H 0 4 L 12/28		G 0 6 F 13/38	3 5 0
G 0 6 F 13/38	3 5 0	H 0 4 L 11/00	3 1 0 Z

審査請求 未請求 請求項の数11 O L (全 23 頁) 最終頁に続く

(21)出願番号 特願平11-361103

(22)出願日 平成11年12月20日 (1999. 12. 20)

(71)出願人 000002369

セイコーエプソン株式会社

東京都新宿区西新宿2丁目4番1号

(72)発明者 松永 浩輔

長野県諏訪市大和3丁目3番5号 セイコ

ーエプソン株式会社内

(72)発明者 金井 裕之

長野県諏訪市大和3丁目3番5号 セイコ

ーエプソン株式会社内

(72)発明者 藤田 信一郎

長野県諏訪市大和3丁目3番5号 セイコ

ーエプソン株式会社内

(74)代理人 100090479

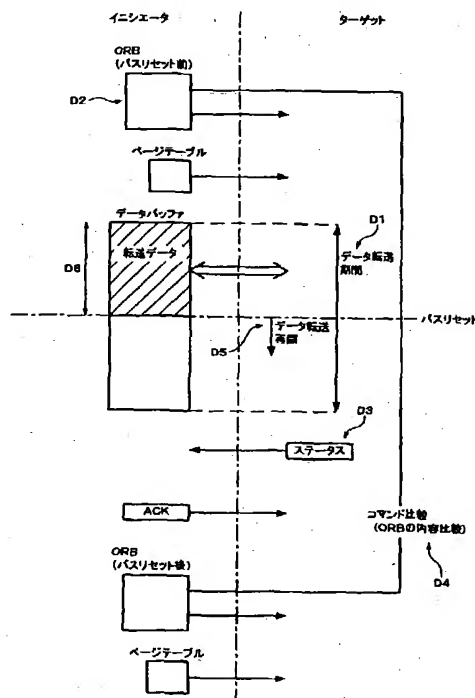
弁理士 井上 一 (外2名)

(54)【発明の名称】 データ転送制御装置、情報記憶媒体及び電子機器

(57)【要約】

【課題】 ノードのトポロジ情報をクリアするリセットが発生した場合に生じる不具合を解消できるデータ転送制御装置、情報記憶媒体及び電子機器の提供。

【解決手段】 IEEE1394規格のデータ転送制御装置において、データ転送期間中にバスリセットが発生し且つバスリセットの前後のORBの内容が同一の場合に、バスリセット発生時点の続きからデータ転送を再開し、プリンタの二重印刷を防止する。バスリセットがデータ転送期間中に発生した場合に継続フラグをオンにする。バスリセット後に最初に転送されてきた、印刷コマンドを含むコマンドブロックORBを、バスリセット前のORBの比較対象として採用する。バスリセットが要因となりイニシエータからACKが返送されない場合に、デッド状態に状態遷移する。スキャナからの転送データの中で、バスリセット発生時点においてイニシエータに未だ転送していない転送データについては、破棄せずに保持しておく。



【特許請求の範囲】

【請求項 1】 バスに接続される複数のノード間でのデータ転送のためのデータ転送制御装置であって、相手ノードとの間でデータ転送が開始してから完了するまでの間のデータ転送期間中に、ノードのトポロジ情報をクリアするリセットが発生したか否かを判断する判断手段と、

該リセットの発生前に相手ノードから転送されてきたデータ転送オペレーション要求のための第 1 のコマンドパケットの内容と、該リセットの発生後に相手ノードから転送されてきたデータ転送オペレーション要求のための第 2 のコマンドパケットの内容とを比較するコマンド比較手段と、

ノードのトポロジ情報をクリアするリセットが前記データ転送期間中に発生したと判断され、且つ、前記第 1、第 2 のコマンドパケットが同一内容であると判断された場合には、リセット発生時点のデータ転送の続きからデータ転送を再開する再開手段と、を含むことを特徴とするデータ転送制御装置。

【請求項 2】 請求項 1 において、

前記判断手段が、

ノードのトポロジ情報をクリアするリセットの発生時点においてデータ転送オペレーション要求のための前記第 1 のコマンドパケットを処理中であり、且つ、該第 1 のコマンドパケットによるデータ転送が既に行われておりデータ転送完了のステータスを相手ノードに転送していない場合に、該リセットが前記データ転送期間中に発生したと判断することを特徴とするデータ転送制御装置。

【請求項 3】 請求項 1 又は 2 において、

前記判断手段が、

ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生したと判断した場合に、データ転送が継続して再開される可能性があることを示す継続フラグをオンにセットすることを特徴とするデータ転送制御装置。

【請求項 4】 請求項 1 乃至 3 のいずれかにおいて、データ転送を再開するアドレスを特定するための情報と、データ転送オペレーション要求のための前記第 1 のコマンドパケットの内容とを、該リセットの発生後、データ転送が再開されるまでの間に記憶するコマンド記憶手段を含むことを特徴とするデータ転送制御装置。

【請求項 5】 請求項 1 乃至 4 のいずれかにおいて、

前記コマンド比較手段が、

ノードのトポロジ情報をクリアするリセットの発生後に相手ノードから転送されてきたコマンドパケットの中で最初に転送されてきたデータ転送オペレーション要求のためのコマンドパケットを、前記第 1 のコマンドパケットの比較対象となる前記第 2 のコマンドパケットとして採用することを特徴とするデータ転送制御装置。

【請求項 6】 請求項 1 乃至 5 のいずれかにおいて、

データ転送完了のステータスを相手ノードに転送したが、ノードのトポロジ情報をクリアするリセットの発生が要因となって相手ノードからアクノリジメントが返って来なかった場合に、データ転送不可状態に状態遷移することを特徴とするデータ転送制御装置。

【請求項 7】 請求項 1 乃至 6 のいずれかにおいて、上層のデバイスから転送されてきた転送データの中で、ノードのトポロジ情報をクリアするリセットの発生時点において相手ノードへ未だ転送していない転送データについては、破棄せずに保持しておくことを特徴とするデータ転送制御装置。

【請求項 8】 請求項 1 乃至 7 のいずれかにおいて、前記リセットが、IEEE 1394 の規格において定義されるバスリセットであることを特徴とするデータ転送制御装置。

【請求項 9】 請求項 1 乃至 8 のいずれかのデータ転送制御装置との間でのデータ転送を制御するためのプログラムを含む情報記憶媒体であって、ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した場合には、該リセットの発生前に転送されたデータ転送オペレーション要求のための第 1 のコマンドパケットと同一内容の第 2 のコマンドパケットを作成し、データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする情報記憶媒体。

【請求項 10】 請求項 1 乃至 8 のいずれかのデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードから受信したデータに所与の処理を施す装置と、処理が施されたデータを出力又は記憶するための装置とを含むことを特徴とする電子機器。

【請求項 11】 請求項 1 乃至 8 のいずれかのデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードに転送するデータに所与の処理を施す装置と、処理が施されるデータを取り込むための装置とを含むことを特徴とする電子機器。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、データ転送制御装置、情報記憶媒体及び電子機器に関し、特に、バスに接続される複数のノード間で IEEE 1394 などの規格に準じたデータ転送を行うためのデータ転送制御装置、情報記憶媒体及び電子機器に関する。

【0002】

【背景技術及び発明が解決しようとする課題】近年、IEEE 1394 と呼ばれるインターフェース規格が脚光を浴びている。この IEEE 1394 は、次世代のマルチメディアにも対応可能な高速シリアルバスインターフェースを規格化したものである。この IEEE 1394 によれば、動画像などのリアルタイム性が要求されるデ

ータも扱うことができる。また、IEEE1394のバスには、プリンタ、スキャナ、CD-RWドライブ、ハードディスクドライブなどのコンピュータの周辺機器のみならず、ビデオカメラ、VTR、TVなどの家庭用電化製品も接続できる。このため、電子機器のデジタル化を飛躍的に促進できるものとして期待されている。

【0003】さて、このIEEE1394においては、バスに電子機器が新たに接続されたり、バスから電子機器が取り外されたりして、バスに接続されるノードが増減すると、いわゆるバスリセットが発生する。そしてバスリセットが発生するとノードのトポロジ情報がクリアされ、その後、トポロジ情報が自動的に再設定される。即ち、バスリセットの発生後、ツリー識別（ルートノードの決定）、自己識別が行われ、その後、アイソクロナスリソースマネージャ等の管理ノードが決定される。そして通常の packets 転送が再開される。

【0004】このようにIEEE1394では、バスリセット後にトポロジ情報が自動的に再設定されるため、いわゆるホット状態でのケーブルの抜き差し（ホットプラグ）が可能となる。このため、一般ユーザは、VTRなどの通常の家庭用電化製品と同じように、電子機器へのケーブルの抜き差しを自由にできるようになり、いわゆるホームネットワークシステムの普及に役立つことができる。

【0005】しかしながら、IEEE1394のバスに接続されたプリンタやスキャナなどのデバイスにおいて、このバスリセットの発生が要因となって以下のような問題が生じることが判明した。

【0006】即ち、IEEE1394のバス上で印刷データの転送中にバスリセットが発生すると、パーソナルコンピュータなどのイニシエータは、印刷データの転送を最初から再度やり直す。従って、ターゲットであるプリンタに対して、印刷データの一部分だけが二重に送られてしまい、二重印刷などの誤印刷の問題が生じる。

【0007】また、スキャナでは、一旦ヘッドが動き出すと、ヘッドを元に戻して再度同じデータを取得することはできない。従って、バスリセットの発生後、イニシエータがデータ転送を最初から再度やり直そうとしても、データ転送を継続することができないという問題がある。

【0008】なお、バスリセットの発生により生じる不具合を解決する従来技術として、例えば特開平11-194902号公報に開示されるものがある。この従来技術では、バスリセットが発生すると、データ処理をホールドし、ネットワーク構成が再構築された後にデータ処理を再開する。

【0009】しかしながら、この従来技術では、バスリセット発生後に転送データを単に再送するだけであり、再送される転送データがバスリセット発生前の転送データの続きか否かの判断については行っていない。従っ

て、この従来技術によっては二重印刷の問題を解決できない。

【0010】本発明は、以上のような技術的課題に鑑みてなされたものであり、その目的とするところは、ノードのトポロジ情報をクリアするリセットが発生した場合に生じる不具合を解消できるデータ転送制御装置、情報記憶媒体及び電子機器を提供することにある。

【0011】

【課題を解決するための手段】上記課題を解決するために本発明は、バスに接続される複数のノード間でのデータ転送のためのデータ転送制御装置であって、相手ノードとの間でデータ転送が開始してから完了するまでの間のデータ転送期間中に、ノードのトポロジ情報をクリアするリセットが発生したか否かを判断する判断手段と、該リセットの発生前に相手ノードから転送されてきたデータ転送オペレーション要求のための第1のコマンドパケットの内容と、該リセットの発生後に相手ノードから転送されてきたデータ転送オペレーション要求のための第2のコマンドパケットの内容とを比較するコマンド比較手段と、ノードのトポロジ情報をクリアするリセットが前記データ転送期間中に発生したと判断され、且つ、前記第1、第2のコマンドパケットが同一内容であると判断された場合には、リセット発生時点のデータ転送の続きからデータ転送を再開する再開手段とを含むことを特徴とする。

【0012】本発明によれば、ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生したか否かが判断される。また、リセット発生前に転送されてきた第1のコマンドパケットの内容と、リセット発生後に転送されてきた第2のコマンドパケットの内容が比較される。そして、データ転送期間中にリセットが発生したと判断され、且つ、第1、第2のコマンドパケットが同一内容であると判断されると、リセット発生時点の続きから（例えばリセット発生時点で転送を完了したデータの次のデータから）、データ転送が再開されるようになる。

【0013】一方、例えば、第1、第2のコマンドパケットが同一内容でないと判断された場合等には、リセット発生後の第2のコマンドパケットは最初から処理されるようになる。

【0014】従って本発明によれば、相手ノードが、リセット発生後にリセット発生前と同一内容のコマンドパケットを転送要求してきた場合には、リセット発生時点の続きからデータ転送を再開できるようになる。従って、例えばデータ転送制御装置の上層のデバイスにデータが重複して転送されてしまい、上層のデバイスが誤動作するなどの問題を解消できる。

【0015】なお、前記判断手段は、ノードのトポロジ情報をクリアするリセットの発生時点においてデータ転送オペレーション要求のための前記第1のコマンドパケ

ットを処理中であり、且つ、該第1のコマンドパケットによるデータ転送が既に行われておりデータ転送完了のステータスを相手ノードに転送していない場合に、該リセットが前記データ転送期間中に発生したと判断することが望ましい。

【0016】また本発明は、前記判断手段が、ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生したと判断した場合に、データ転送が継続して再開される可能性があることを示す継続フラグをオンにセットすることを特徴とする。このようにすれば、継続フラグがオフの場合には、コマンドパケットの比較処理が行われないようにすることができる。従って、無用なコマンドパケットの内容比較処理が行われないようになり、処理負担の軽減化を図れる。

【0017】また本発明は、データ転送を再開するアドレスを特定するための情報と、データ転送オペレーション要求のための前記第1のコマンドパケットの内容とを、該リセットの発生後、データ転送が再開されるまでの間に記憶するコマンド記憶手段を含むことを特徴とする。このような情報を記憶しておけば、データ転送の再開処理を簡素な処理で実現できるようになる。

【0018】また本発明は、前記コマンド比較手段が、ノードのトポロジ情報をクリアするリセットの発生後に相手ノードから転送されてきたコマンドパケットの中で最初に転送されてきたデータ転送オペレーション要求のためのコマンドパケットを、前記第1のコマンドパケットの比較対象となる前記第2のコマンドパケットとして採用することを特徴とする。このようにすれば、最初のデータ転送オペレーション要求のためのコマンドパケットが転送されてくるまで、コマンドパケットの内容比較処理が実行されるのが、繰り越されるようになる。これにより、無用なコマンドパケットの内容比較処理が行われるのを防止でき、処理負担の軽減化を図れる。

【0019】また本発明は、データ転送完了のステータスを相手ノードに転送したが、ノードのトポロジ情報をクリアするリセットの発生が要因となって相手ノードからアクノリッジメントが返って来なかった場合に、データ転送不可状態に状態遷移することを特徴とする。このように、相手ノードからアクノリッジメントが返って来なかった場合には、相手ノードがステータスを受け取ったか否かが不明となる。従って、このような場合にリセット発生時点の続きからデータ転送を開始してしまうと、誤ったデータ転送が行われてしまう可能性がある。本発明によれば、ノードのトポロジ情報をクリアするリセットが要因となって相手ノードからアクノリッジメントが返って来なかった場合には、データ転送不可状態に状態遷移するため、このような誤ったデータ転送が行われる事態を防止できる。

【0020】また本発明は、上層のデバイスから転送されてきた転送データの中で、ノードのトポロジ情報をク

リアするリセットの発生時点において相手ノードへ未だ転送していない転送データについては、破棄せずに保持しておくことを特徴とする。このようにすれば、スキナ等により取り込んだデータが、リセットの発生により失われてしまう等の不具合を防止できる。

【0021】なお本発明では、前記リセットが、IEEE1394の規格において定義されるバスリセットであることが望ましい。

【0022】また本発明は、上記のいずれかのデータ転送制御装置との中でのデータ転送を制御するためのプログラムを含む情報記憶媒体であって、ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した場合には、該リセットの発生前に転送されたデータ転送オペレーション要求のための第1のコマンドパケットと同一内容の第2のコマンドパケットを作成し、データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする。このようにすれば、誤ったデータ転送再開処理が行われてしまう事態を防止でき、リセットの発生が要因となる不具合の発生を防止できる。

【0023】また本発明に係る電子機器は、上記のいずれかのデータ転送制御装置と、前記データ転送制御装置及びバスを介して相手ノードから受信したデータに所与の処理を施す装置と、処理が施されたデータを出力又は記憶するための装置とを含むことを特徴とする。また本発明に係る電子機器は、上記のいずれかのデータ転送制御装置と、前記データ転送制御装置及びバスを介して相手ノードに送信するデータに所与の処理を施す装置と、処理が施されるデータを取り込むための装置とを含むことを特徴とする。

【0024】本発明によれば、ノードのトポロジ情報をクリアするリセットの発生によりシステムに不具合が生じる事態を防止でき、電子機器が誤動作するのを防止できる。またデータ転送の高速化を図れ、電子機器の低コスト化、電子機器の処理の高速化なども図ることができる。

【0025】

【発明の実施の形態】以下、本発明の好適な実施形態について図面を用いて詳細に説明する。

【0026】1. IEEE1394

まず、IEEE1394について簡単に説明する。

【0027】1. 1 概要

IEEE1394 (IEEE1394-1995、P1394. a) では100~400Mbpsの高速なデータ転送が可能となっている (P1394. bでは800~3200Mbps)。また、転送速度が異なるノードをバスに接続することも許される。

【0028】各ノードはツリー状に接続されており、1つのバスに最大で63個のノードが接続可能になっている。なお、バスブリッジを利用すれば約64000個のノードを接続することも可能である。

【0029】IEEE1394では、パケットの転送方式として非同期転送とアイソクロナス転送が用意されている。ここで非同期転送は、信頼性が要求されるデータの転送に好適な転送方式であり、アイソクロナス転送は、リアルタイム性が要求される動画像や音声などのデータの転送に好適な転送方式である。

【0030】1. 2 層構造

IEEE1394の層構造（プロトコル構成）を図1に示す。

【0031】IEEE1394のプロトコルは、トランザクション層、リンク層、物理層により構成される。また、シリアルバスマネジメントは、トランザクション層、リンク層、物理層をモニターしたり制御したりするものであり、ノードの制御やバスのリソース管理のための種々の機能を提供する。

【0032】トランザクション層は、上位層にトランザクション単位のインターフェース（サービス）を提供し、下層のリンク層が提供するインターフェースを通して、リードトランザクション、ライトトランザクション、ロックトランザクション等のトランザクションを実施する。

【0033】ここで、リードトランザクションでは、応答ノードから要求ノードにデータが転送される。一方、ライトトランザクションでは、要求ノードから応答ノードにデータが転送される。またロックトランザクションでは、要求ノードから応答ノードにデータが転送され、応答ノードがそのデータに処理を施して要求ノードに返信する。

【0034】リンク層は、アドレッシング、データチェック、パケット送受信のためのデータフレーミング、アイソクロナス転送のためのサイクル制御などを提供する。

【0035】物理層は、リンク層により使用されるロジカルシンボルの電気信号への変換や、バスの調停や、バスの物理的インターフェースを提供する。

【0036】1. 3 SBP-2

さて、図2に示すように、IEEE1394のトランザクション層の一部の機能を含む上位のプロトコルとして、SBP-2 (Serial Bus Protocol-2) と呼ばれるプロトコルが提案されている。

【0037】ここでSBP-2は、SCSIのコマンドセットをIEEE1394のプロトコル上で利用可能にするために提案されたものである。このSBP-2を用いれば、既存のSCSI規格の電子機器で使用されていたSCSIのコマンドセットに最小限の変更を加えて、IEEE1394規格の電子機器に使用できるようになる。従って、電子機器の設計や開発を容易化できる。また、SCSIのコマンドだけではなく、デバイス固有のコマンドもカプセル化して利用できるように、非常に汎用性が高い。

【0038】図3に示すようにSBP-2では、まず、イニシエータ（例えばパーソナルコンピュータ）により作成されたログインORB (Operation Request Block) を用いてログイン処理が行われる（ステップT1）。次に、ダミーORBを用いてフェッチエージェントの初期化が行われる（ステップT2）。そして、コマンドブロックORB（ノーマルコマンドORB）を用いてコマンド処理が行われ（ステップT3）、最後に、ログアウトORBを用いてログアウト処理が行われる（ステップT4）。

【0039】ここで、ステップT3のコマンド処理においては、図4のA1に示すように、イニシエータがライト要求パケットを転送して（ライト要求トランザクションを発行して）、ターゲットのドアベルレジスタをリングする。すると、A2に示すように、ターゲットがリード要求パケットを転送し、イニシエータが対応するリード応答パケットを返す。これにより、イニシエータが作成したORB（コマンドブロックORB）が、ターゲットのデータバッファにフェッチされる。そして、ターゲットは、フェッチされたORBに含まれるコマンドを解析する。

【0040】そして、ORBに含まれるコマンドがSCSIのライトコマンドであった場合には、A3に示すように、ターゲットがリード要求パケットをイニシエータに転送し、イニシエータが対応するリード応答パケットを返す。これにより、イニシエータのデータバッファに格納されているデータがターゲットに転送される。そして、例えばターゲットがプリンタであった場合には、転送されたデータがプリンタエンジンにより印刷される。

【0041】一方、ORBに含まれるコマンドがSCSIのリードコマンドであった場合には、図5のB1に示すように、ターゲットは、一連のライト要求パケットをイニシエータに転送する。これにより、例えばターゲットがスキャナであった場合には、スキャナエンジンにより取得されたスキャンデータが、イニシエータのデータバッファに転送されることになる。

【0042】このSBP-2によれば、ターゲットは、自身が都合の良いときに要求パケットを転送して（トランザクションを発行して）、データを送受信できる。従って、イニシエータとターゲットが同期して動く必要がなくなるため、データ転送効率を高めることができる。

【0043】なお、IEEE1394の上位プロトコルとしては、SBP-2以外にも、FCP (Function Control Protocol) と呼ばれるプロトコルなども提案されている。

【0044】さて、ターゲット、イニシエータ間でデータ転送を行う場合、図6 (A) のようにイニシエータ（相手ノード）のデータバッファ（記憶手段）にページテーブルが存在する場合と、存在しない場合がある。

【0045】そして、ページテーブルが存在する場合に

は、図6(B)に示すように、イニシエータが作成したORBの中には、そのページテーブルのアドレスやエレメント数が含まれる。そして、転送データのアドレス（読み出しアドレス、書き込みアドレス）は、このページテーブルを用いて間接アドレス指定される。

【0046】一方、ページテーブルが存在しない場合には、図6(C)に示すように、ORBの中にアドレスとデータ長が含まれ、転送データのアドレスが直接アドレス指定される。

【0047】1. 4 バスリセット

IEEE1394では、電源が投入されたり、途中でデバイスの抜き差しが発生すると、バスリセットが発生する。即ち、各ノードは、ポートの電圧変化を監視している。そして、バスに新たなノードが接続されるなどしてポートの電圧に変化が生じると、この変化を検出したノードは、バス上の他のノードに対して、バスリセットが発生したことを知らせる。また、各ノードの物理層は、バスリセットが発生したことをリンク層に伝える。

【0048】そして、このようにバスリセットが発生すると、ノードIDなどのトポロジ情報がクリアされる。そして、その後、トポロジ情報が自動的に再設定される。即ち、バスリセット後、ツリー識別、自己識別が行われる。その後、アイソクロナスリソースマネージャ、サイクルマスタ、バスマネージャ等の管理ノードが決定される。そして、通常の packets 転送が再開される。

【0049】このようにIEEE1394では、バスリセット後にトポロジ情報が自動的に再設定されるため、電子機器のケーブルを自由に抜き差しできるようになり、いわゆるホットプラグを実現できる。

【0050】なお、トランザクションの途中でバスリセットが発生した場合には、そのトランザクションはキャンセルされる。そして、キャンセルされたトランザクションを発行した要求ノードは、トポロジ情報が再設定された後に、要求 packets を再度転送する。また、応答ノードは、バスリセットによりキャンセルされたトランザクションの応答 packets を要求ノードに返送してはならない。

【0051】2. 全体構成

次に、本実施形態のデータ転送制御装置の全体構成例について図7を用いて説明する。なお、以下では、イニシエータとの間でデータ転送を行うターゲットがプリンタである場合について例にとり説明するが、本発明はこれに限定されない。

【0052】本実施形態のデータ転送制御装置10は、PHYデバイス12（物理層のデバイス）、リンクデバイス14（リンク層のデバイス）、CPU16（プロセッサ）、データバッファ18（記憶手段）、ファームウェア20（処理手段）を含む。なお、PHYデバイス12、リンクデバイス14、CPU16、データバッファ18は、任意の構成要素であり、本実施形態のデータ転

送制御装置10は、これらの構成要素を全て含む必要はない。

【0053】PHYデバイス12は、図1の物理層のプロトコルをハードウェアにより実現するための回路であり、リンクデバイス14により使用されるロジカルシンボルを電気信号に変換する機能を有する。

【0054】リンクデバイス14は、図1のリンク層のプロトコルやトランザクション層のプロトコルの一部をハードウェアにより実現するための回路であり、ノード間での packets 転送のための各種サービスを提供する。

【0055】CPU16は、装置全体の制御やデータ転送の制御を行うものである。

【0056】データバッファ18は、転送データ（packets）を一時的に格納するバッファであり、SRAM、SDRAM、或いはDRAMなどのハードウェアにより構成される。なお、本実施形態では、データバッファ18は、ランダムアクセス可能な packets 記憶手段として機能する。

【0057】ファームウェア20は、CPU16上で動作する種々の処理ルーチン（処理モジュール）を含むプログラムであり、トランザクション層のプロトコルは、このファームウェア20と、ハードウェアであるCPU16等により実現される。

【0058】なお、イニシエータであるパーソナルコンピュータ100が含むデバイスドライバ102は、周辺機器を管理制御するための種々の処理ルーチンを含むプログラムである。このプログラムは、情報記憶媒体110（FD、CD-ROM、DVD、ROM）を利用してパーソナルコンピュータ100にインストールされる。

【0059】ここで、デバイスドライバ102のプログラムは、ホストシステムが有する情報記憶媒体（ハードディスク、磁気テープ等）からインターネットなどのネットワークを介してダウンロードし、パーソナルコンピュータ100にインストールするようにしてもよい。このようなホストシステムが有する情報記憶媒体の使用も本発明の範囲内に含まれる。

【0060】ファームウェア20（F/W）は、コミュニケーション部30（COM）、マネージメント部40（MNG）、プリントタスク部50（PRT）、フェッチ部60（FCH）を含む。

【0061】ここで、コミュニケーション部30は、リンクデバイス14などのハードウェアとの間のインターフェースとして機能する処理モジュールである。

【0062】マネージメント部40（マネージメントエージェント）は、ログイン、リコネクト、ログアウト、リセット等の管理を行う処理モジュールである。例えばイニシエータがターゲットにログインを要求した場合には、まず、このマネージメント部40が、このログイン要求を受け付けることになる。

【0063】プリントタスク部50は、後段のアプリケ

ーション層（上層）であるプリンタエンジンとの間のデータ転送処理を行う処理モジュールである。

【0064】フェッチ部60（フェッチエージェント、コマンドブロックエージェント）は、コマンドブロックORBを含むコマンドを実行するための処理モジュールである。フェッチ部60は、単一の要求しか扱うことができないマネージメント部40と異なり、イニシエータからの要求により自身がフェッチしたORBのリンクリストも扱うことができる。

【0065】フェッチ部60は、判断部62、コマンド記憶部64、コマンド比較部66、アドレス記憶部68、アドレス比較部70、データ転送再開部72を含む。

【0066】ここで判断部62は、イニシエータ（相手ノード）との間で印刷データを転送するデータ転送期間中に、バスリセット（広義には、ノードのトポロジ情報をクリアするリセット）が発生したか否かを判断する処理を行う。

【0067】コマンド記憶部64は、バスリセットの発生前にイニシエータから転送されてきたORB（コマンドブロックORB。広義には、データ転送オペレーション要求のためのコマンドパケット）の内容を、バスリセットが発生した時点やリコネクトが成功した時点などで記憶するための処理を行う。

【0068】コマンド比較部66は、バスリセットの発生前にイニシエータから転送されてきたORB（コマンドブロックORB）の内容（コマンド記憶部64により記憶された内容）と、バスリセットの発生後にイニシエータから転送されてきたORBの内容とを比較する処理を行う。

【0069】アドレス記憶部68は、イニシエータとの間で転送される転送データ（印刷データ）の先頭アドレス（第1のアドレス）を記憶するための処理を行う。

【0070】アドレス比較部70は、バスリセットが発生した場合に、アドレス記憶部68により記憶された先頭アドレス（第1のアドレス）と、バスリセット発生後の転送データの先頭アドレス（第2のアドレス）とを比較する処理を行う。

【0071】データ転送再開部72は、バスリセットがデータ転送期間中に発生したと判断され、且つ、ORB（コマンドブロックORB）の内容が一致したと判断された場合に、バスリセット発生時点のデータ転送の続き（バスリセット発生時点で転送したデータの次のデータ）からデータ転送を再開する処理を行う。

【0072】3. 処理の概要

次に、本実施形態の処理の概要について説明する。

【0073】図8は、ターゲット側（ファームウェア）の処理の概要について示すフローチャートである。

【0074】イニシエータから印刷要求があると、ターゲットは、イニシエータのデータバッファからORBを

リードする（ステップS1）。そして、ページテーブルが存在する場合には、ORBに含まれるページテーブルアドレス（図6（B）参照）に基づいて、イニシエータのデータバッファからページテーブルをリードする（ステップS2）。次に、リードしたページテーブルに基づいてイニシエータのデータバッファから印刷データをリードする（ステップS3）。そして、ページテーブルにより指定される印刷データを全てリードすると、ステータスをライトして、データ転送が成功したか否かなどのステータスをイニシエータに伝える（ステップS4）。以上の処理を、全ての印刷データが転送されるまで繰り返す（ステップS5）。

【0075】そして本実施形態では、印刷データの転送中（データ転送期間）にバスリセットが発生すると、リコネクト後の最初の印刷要求時に以下の処理を行う。

【0076】即ち、まず、バスリセット前のORBの内容や印刷データの先頭アドレスと、バスリセット後のORBの内容や印刷データの先頭アドレスとが同一か否かを判断する（ステップS6）。そして、同一であると判断した場合には、バスリセット発生時点の続きからデータ転送を再開する（ステップS7）。一方、同一でないと判断した場合には、バスリセット後のORBを新規のORBとして最初から処理する（ステップS8）。

【0077】図9は、イニシエータ側（デバイスドライバ）の処理の概要について示すフローチャートである。

【0078】アプリケーションプログラムからの印刷ジョブが発生すると、イニシエータは、印刷のためのORBやページテーブルを作成し、データバッファに書き込む（ステップS10）。次に、作成したORBをリードするようにターゲットに対して指示する（ステップS11。図4のA1参照）。

【0079】次に、バスリセットが発生したか否かを判断し（ステップS12）、発生しなかった場合には、ステータスがターゲットから送られてきたか否かを判断する（ステップS13）。そして、送られてきた場合には、全ての印刷データが転送されたか否かを判断し（ステップS14）、転送されていない場合には、ステップS10に戻り、転送された場合には印刷ジョブを終了する。

【0080】そして本実施形態では、ステップS12でバスリセットが発生したと判断されると、イニシエータが、ORB、ページテーブルを再作成し（ステップS15）、再作成したORBをリードするようにターゲットに対して指示する（ステップS11）。この場合にイニシエータ（デバイスドライバ）は、バスリセット発生前のORBの内容や印刷データの先頭アドレスと、バスリセット発生後のORBの内容や印刷データの先頭アドレスとが同一になるように、ORBを再作成する。

【0081】4. 本実施形態の特徴

さて、印刷データの転送中にバスリセットが発生する

と、以下のような問題が生じることが判明した。

【0082】例えば図10(A)に示すように、C1に示す位置(アドレス)までデータを転送したところで、バスリセットが発生したとする。この場合には、バスリセット発生時点で処理中であったトランザクションは全てキャンセルされる。従って、バスリセット前に印刷データの転送を要求していたイニシエータは、図10

(B)に示すように、バスリセット後に印刷のためのORBを再度作成して、印刷データの転送を最初からやり直すようにターゲットに指示する。このため、図10

(B)のC2に示す位置からデータ転送が再開されてしまい、印刷データの一部分だけが二重に送られてしまう。この結果、図10(C)に示すような二重印刷の問題が発生する。

【0083】このような問題を解決するために、本実施形態では、以下に説明するような手法を採用している。

【0084】即ち本実施形態では、まず、図11のD1に示すように、データ転送期間中(イニシエータとの間で印刷データの転送が開始してから完了するまでの期間)にバスリセットが発生したか否かを判断する。より具体的には、バスリセットの発生時点において、D2に示すORB(コマンドブロックORB)を処理中であり、且つ、そのORBによるデータ転送が既に行われており、D3に示すデータ転送完了のステータスをイニシエータに転送していない場合に、バスリセットがデータ転送期間中に発生したと判断する。

【0085】また本実施形態では、図11のD4に示すように、バスリセットの発生後にイニシエータがリコネクに成功し、新たなORBを作成して転送要求してきた場合に、バスリセット前のORB(第1のコマンドパケット)の内容とバスリセット後のORB(第2のコマンドパケット)の内容を比較する。

【0086】そして、データ転送期間中にバスリセットが発生したと判断され、且つ、バスリセット前のORBの内容とバスリセット後のORBの内容が同一内容と判断された場合には、D5に示すように、バスリセット発生時点のデータ転送の続きからデータ転送を再開する

(図8のステップS7参照)。即ち、バスリセット発生時点で既に転送を完了していたデータの次のデータからデータ転送を再開する。

【0087】一方、データ転送期間中にバスリセットが発生しなかった場合や、バスリセットの前と後でORBの内容が同一でない場合には、バスリセット後のORBを新規のものとして最初から処理する(図8のステップS8参照)。

【0088】このようにすることで、図11のD6に示す部分の転送データが、図10(B)の場合と異なり、二重転送されないようになる。従って、図10(C)に示すような誤印刷が生じなくなる。また二重転送を避けることができるため、転送時間も短縮できる。

【0089】例えば、データ転送期間以外の期間にバスリセットが発生した場合には、D5のようにバスリセット発生時点の続きからデータ転送を再開しなくても、転送データが二重転送されることはないため、問題はない。また、データ転送期間以外の期間にバスリセットが発生した場合には、データ転送の再開処理を行うよりも、ORBを最初から処理した方が、処理も簡素になり、処理負担も軽くなる。

【0090】また、本実施形態と異なる手法として、ORBの内容の比較処理を行わずに、常に、バスリセット発生時点の続きからデータ転送を再開するという手法も考えられる。

【0091】しかしながら、この手法によると、例えば、バスリセット後にイニシエータが印刷データの転送処理をキャンセルし、バスリセット前と全く異なるORBを作成した場合にも、図11のD5からデータ転送が再開されてしまうという不具合が生じる。

【0092】これに対して本実施形態では、ORBの内容がバスリセットの前後で同一の場合には、図11のD5からデータ転送が再開するが、同一でない場合には、全く新規のORBとして処理されるため、上記のような不具合が生じない。

【0093】なお本実施形態では、ORBの内容比較の際に種々の情報を比較している。例えば図12に示すように、本実施形態では、コマンドブロックORBが含むページテーブル存在フラグPや、データサイズや、コマンドブロック(コマンドセット)フィールドの中のオペレーションコード(印刷コマンド、リードコマンドなどを区別するコード)やデータ長を比較している。またORBが、ORBを識別するための識別情報(例えば順序番号)を含む場合には、この識別情報も比較している。このような情報を比較することで、バスリセット前後のORBが同一か否かを簡素な処理で確実に判断できるようになる。

【0094】さて本実施形態では、図13のE1に示すように、バスリセットがデータ転送期間中に発生したと判断した場合には、データ転送が継続して再開される可能性があることを示す継続フラグをオンにセットするようにしている。例えば、バスリセットの発生後、リコネクが成功した時などに、バスリセットがデータ転送期間中に発生したか否かが判断され、発生したと判断された場合に継続フラグがオンにセットされる。

【0095】そして図13のE2に示すように、ORBの内容比較は、この継続ビットがオンであることを条件に行われる。即ち、継続フラグがオフの場合には、ORBの内容比較は行われない。これにより、無用なORBの内容比較処理が行われることが防止されるようになり、ファームウェアの処理負担を軽減化できる。

【0096】また本実施形態では、データ転送を再開するアドレス(図11のD5)を特定するための情報や、

バスリセット前のORBの内容などを、バスリセット発生時点やリコネクトが成功した時点（広義には、バスリセットの発生後、データ転送が再開されるまでの間）に記憶している。このような情報を記憶しておくことで、バスリセット後にイニシエータからORBが再度転送されてきた時に、バスリセット発生時点の続きからデータ転送を再開するか否かの判断を容易に行えるようになる。またデータ転送の再開処理も簡素化できるようになる。

【0097】なお、データ転送を再開するアドレスを特定するための情報としては、具体的には次のような情報を考えることができる。

【0098】例えば図14に示すように、バスリセット発生時点において、ページテーブルのN番目のセグメントSEGNが指定する転送データを転送していたとする。この場合には、データ転送を再開するアドレス（F1に示すアドレス）を特定するための情報としては、ページテーブルの先頭セグメントSEG1に格納されるアドレス（転送データの先頭アドレス）や、セグメントSEG1～SEG（N-1）により転送されたデータのサイズDS1や、セグメントNのセグメント番号や、バスリセット発生時点までにセグメントNにより転送されたデータのサイズDS2などを考えることができる。

【0099】また本実施形態では、バスリセット発生後にイニシエータから転送されてきたORBの中で、最初に転送されてきた印刷コマンドを含むORB（コマンドブロックORB）を、バスリセット前のORBの比較対象となるORBとして採用するようにしている。

【0100】例えば図15に示すように、バスリセット後に、印刷コマンドを含むコマンドブロックORBが直ぐに転送されてくるとは限らない。即ち、イニシエータが、印刷コマンドを含むコマンドブロックORBを転送要求する前に、ダミーORBやマネージメントORBを転送要求してくる場合がある。或いは、プリンタのステータス等を調べるためにリードコマンドを含むコマンドブロックORBを転送要求してくる場合もある。

【0101】本実施形態では図15のG1に示すように、ダミーORB等が転送されてきても、ORBの内容比較処理を行わず、印刷コマンドを含むコマンドブロックORBが転送されてきた時に初めてORBの内容比較処理を行う。即ち、印刷コマンドを含む最初のコマンドブロックORBが来るまで、ORBの内容比較処理の実行を繰り越す（順送りにする）。これにより、無用なORBの内容比較処理が行われるのが防止され、ファームウェアの処理負担を軽減化できる。

【0102】さて、バスリセットの発生時期は全くの任意である。従って、例えば図16に示すように、ターゲットがデータ転送完了のステータスをイニシエータに転送したが、バスリセットの発生が要因となってイニシエータからACK（ACKコンプリート）が返って来

ず、ACKミッシングになる場合がある。

【0103】このような場合には、バスリセットの発生が要因となってイニシエータがステータスを受け取れず、ACKミッシングになったという第1のケースと、イニシエータはステータスを受け取り、ACKを返したが、バスリセットの発生が要因となってACKミッシングになったという第2のケースが考えられる。

【0104】そして、上記第1のケースでは、イニシエータはデータ転送が不成功であったと考え、バスリセット後に同一のORBを再度作成するという第1の処理を行う。一方、上記第2のケースでは、イニシエータはデータ転送が成功したと考え、バスリセット後に次のORBを作成するという第2の処理を行う。

【0105】ところが、ターゲットにはACKミッシングであったという情報しか伝わらないため、ターゲットは、イニシエータが上記第1、第2の処理のいずれを行ったのかを知ることができない。従ってこのような場合にバスリセット発生時点の続きからデータ転送を開始してしまうと、誤ったデータ転送が行われてしまう可能性がある。

【0106】そこで本実施形態では図16に示すように、バスリセットの発生が要因となってイニシエータからアクノリッジメントが返って来なかった場合には、デッド状態（データ転送不可状態）に移移する。このようにすることで、誤ったデータ転送が行われる事態を防止できる。

【0107】さて、以上では、ターゲットがプリンタである場合について主に説明したが、ターゲットがスキャナである場合には以下のような問題がある。

【0108】即ち、スキャナでは、一旦ヘッドが動き出すと、ヘッドを元に戻して再度同じデータを取得（スキャン）することはできないという問題がある。

【0109】そこで本実施形態では図17に示すように、スキャナエンジン（広義には上層のデバイス）からの転送データの中で、バスリセットの発生時点においてイニシエータに未だ転送していない転送データについては、破棄せずに保持するようにしている。即ち図17のH1に示すように、スキャナエンジンから取り込んでデータバッファに格納したが、IEEE1394のバスを介してイニシエータに未だ転送していないデータについては、データバッファをクリアしないようにして破棄せずに保持しておく。

【0110】このようにすれば、バスリセット発生後に図11に示す手法でバスリセット発生時点の続きからデータ転送を再開した場合にも、バスリセット発生前に取り込んだ画像データが失われてしまう等の不具合を防止できる。

【0111】5. 詳細な処理例

次に本実施形態の詳細な処理例について図18～図22のフローチャートを用いて説明する。

【0112】図18～図20は、バスリセット発生時（リコネクト時）の処理の詳細例を示すフローチャートである。

【0113】バスリセットが発生すると、ターゲットは、まず、イニシエータがログインしているか否かを判断し（ステップS20）、ログインしている場合にはIEEE1394のバス上の全ての転送処理（トランザクション）をキャンセルする（ステップS21）。一方、ログインしていない場合には、バスリセットが発生しても特別な処理は不要であるため、何もしない（ステップS22）。

【0114】次に、既にバスリセット処理に入っているか否かを判断する（ステップS23）。これにより、バスリセットが複数回発生した場合にそれに対応するバスリセット処理が無用に複数回繰り返される事態を防止できる。

【0115】次に、バスリセット発生時点のACK（アクノリジメント）の状態を記憶しておく（ステップS24）。これにより、その後に発生するトランザクション（例えばリコネクトのトランザクション）により、バスリセット直後のACKの内容が消されてしまう事態を防止できる。

【0116】次に、IEEE1394のバス上で転送済みのデータのサイズ（バイト数）を記憶する（ステップS25）。即ち、バスリセット発生時点において処理中であったセグメントでの転送済みのデータのサイズ（図14のDS2）を記憶する。そして、ステップS23の判断のために、バスリセット処理中であることを示すフラグをオンにする（ステップS26）。即ち、このフラグがオンになると、その後に、バスリセットが発生しても、ステップS24～S26の処理は行われない。

【0117】次に、イニシエータからのリコネクト待ちになり（ステップS27）、イニシエータによりリコネクトされたか否かを判断する（ステップS28）。そして、リコネクトされなかった場合には、ログインORBのリコネクトフィールドで指定されたリコネクトタイムアウト時間が経過したか否かを判断する（ステップS29）。そして、経過した場合には、図13で説明した継続フラグ（データ転送が継続して再開される可能性があることを示すフラグ）をオフにして（ステップS30）、ログアウト状態に状態遷移する（ステップS31）。

【0118】一方、リコネクトタイムアウト時間内にリコネクトされた場合には、リコネクトしてきたイニシエータが、バスリセット前にログインしていたイニシエータか否かを判断し（ステップS32）、バスリセット前と異なるイニシエータであった場合には、そのイニシエータのリコネクトを拒否し、リコネクト待ちに戻る。

【0119】バスリセット前と同じイニシエータがログインしてきた場合には、印刷のためのコマンドブロック

ORB（印刷コマンドを含むORB）を、バスリセット発生時点において処理中だったか否かを判断する（図19のステップS33）。そして、処理中でなかった場合には、継続フラグをオフにして（ステップS36）、アイドル状態に状態遷移する（ステップS37）。

【0120】一方、印刷のためのコマンドブロックORBを処理中であった場合には、ステータスのライト中（ステータスをライトしてからACKが返ってくるまでの期間）にバスリセットが発生したか否かを判断する（ステップS34）。そして、ステータスのライト中にバスリセットが発生した場合には、図18のステップS24で記憶したACKの情報に基づいて、ACKコンプリートか否かを判断する（ステップS35）。

【0121】そして、ACKコンプリートの場合には、継続フラグをオフにして（ステップS36）、アイドル状態に状態遷移する（ステップS37）。一方、ACKコンプリートでない場合には、ACKミッシングか否かを判断する（ステップS38）。そして、ACKミッシングでなければ何れもせず（ステップS39）、ACKミッシングの場合には、継続フラグをオフにして（ステップS40）、図16で説明したようにデッド状態（データ転送不可状態）に状態遷移する（ステップS41）。

【0122】ステップS34でバスリセットの発生がステータスのライト中でないと判断された場合には、処理中のORBの転送（印刷）データを1バイトでも後段のプリンタエンジンに転送したか否かを判断する（ステップS42）。そして、1バイトも転送していなかった場合には継続フラグをオフにし（ステップS43）、アイドル状態に状態遷移する（ステップS49）。

【0123】一方、1バイトでもプリンタエンジンに転送していた場合には、ORBの内容（データサイズ、ページテーブル存在フラグP、コマンドブロック）や、バスリセット発生時点までに転送できたデータのサイズを記憶する（ステップS44）。このデータのサイズは、バスリセット発生時点で後段のプリンタエンジンに既に転送したデータのバイト数と、バスリセット発生時点でIEEE1394のバス上でのデータ転送は既に完了し、後段のプリンタエンジンに転送中又はこれから転送する予定のデータのバイト数の合計に相当する。即ち、例えば、プリンタで既に印刷したデータのバイト数と、プリンタで現在印刷中又はこれから印刷予定のデータのバイト数の合計に相当する。

【0124】次に、ページテーブルが存在するか否かを判断し（ステップS45）、存在しない場合には、ORBのデータスクリプタの内容を記憶する（ステップS46）。即ち、ページテーブルが存在しない場合には、直接アドレス指定の場合の転送データのアドレス及びデータ長が記憶される（図6（C）参照）。

【0125】一方、ページテーブルが存在する場合には、図14で説明したように、ページテーブルの先頭セ

グメントの内容（アドレス、データ長）と、バスリセット発生時点で処理中だったセグメントの内容（アドレス、データ長）及びセグメント番号を記憶する（ステップS47）。そして、図13で説明した継続フラグをオンにして（ステップS48）、アイドル状態に状態遷移する（ステップS49）。

【0126】このように本実施形態では、データ転送期間中にバスリセットが発生したと判断した場合に（図19のステップS33、S34、図20のステップS42）、継続フラグがオンになる（ステップS48）。

【0127】図21、図22は、通常時の処理の詳細例を示すフローチャートである。

【0128】まず、イニシエータからORBのリードを指示されたか否か（ドアベルレジスタがリングされたか否か）を判断し（ステップS51）、指示されなかった場合にはアイドル状態にとどまる（ステップS50）。

一方、指示された場合には、イニシエータが作成したORBをイニシエータからリードする（ステップS52）。そして、ORBが含むページテーブル存在フラグに基づいて、ページテーブルが存在するか否かを判断する（ステップS53）。そして、ページテーブルが存在する場合には、ページテーブルのセグメントを例えば8セグメントずつリードする（ステップS54）。

【0129】次に、ORBのコマンドブロックにあるオペレーションコードに基づいて、図15で説明したように、リードしたORBが印刷のためのコマンドブロックORBか否かを判断する（ステップS55）。そして、印刷のためのコマンドブロックORBであった場合には、ステップS54でリードした8セグメントが、ページテーブルの最初の8セグメント（先頭セグメントを含む8セグメント）か否かを判断し（ステップS56）、最初の8セグメントであった場合には、図22に示すコマンド・アドレス比較処理に移行する（ステップS57）。

【0130】ステップS55で印刷のためのコマンドブロックORBではないと判断された場合、ステップS56で最初の8セグメントでないと判断された場合、及び、ステップS57のコマンド・アドレス比較処理が終了した場合には、データのリード/ライトを行う（ステップS58）。そして、1セグメント分のデータ、8セグメント分のデータをリード/ライトするまで繰り返す（ステップS59、S60）。

【0131】次に、ページテーブルの全てのセグメントをリード/ライトしたか否かを判断し（ステップS61）、全てのセグメントをリード/ライトしていない場合には、ページテーブルの次の8セグメントをリードする（ステップS54）。一方、ページテーブルの全てのセグメントをリードした場合には、イニシエータに対してステータスをライトする（ステップS62）。そして、印刷物の印刷のための全てのORBをリードしたか

否かを判断し（ステップS63）、次のORBがある場合にはステップS52に戻り、次のORBが無い場合にはアイドル状態に状態遷移する（ステップS50）。

【0132】ステップS53でページテーブルが存在しないと判断した場合には、リードしたORBが印刷のためのコマンドブロックORBか否かを判断する（ステップS64）。そして、印刷のためのコマンドブロックORBであった場合には、図22に示すコマンド・アドレス比較処理に移行する（ステップS65）。

【0133】一方、印刷のためのコマンドブロックORBではないと判断された場合、及び、コマンド・アドレス比較処理が終了した場合には、データをリード/ライトし（ステップS66）、全てのデータをリード/ライトするまで繰り返す（ステップS67）。そして、全てのデータをリード/ライトした場合には、ステップS62に移行し、イニシエータに対してステータスをライトする。

【0134】図22のコマンド・アドレス比較処理においては、まず、継続フラグがオンか否かを判断する（ステップS70）。この継続フラグは、図20のステップS48においてオンにされるフラグである。そして、継続フラグがオフの場合にはステップS76に移行して、転送データの先頭アドレス（ページテーブルの先頭セグメントのアドレス）を記憶し、コマンド・アドレス比較処理を終了する。

【0135】継続フラグがオンの場合には、図11、図12で説明したように、リードしたORBの内容がバスリセット前のORBの内容と同一か否かを判断する（ステップS71）。この場合に、比較対象となるバスリセット前のORBの内容は、図20のステップS44において記憶されている。また、本実施形態では、アドレス比較（ステップS72）に先立って、ORBの内容比較（ステップS71）を行っている。

【0136】ORBの内容がバスリセット前と同一であった場合には、転送データの先頭アドレスがバスリセット前と同一か否かを判断する（ステップS72）。そして、同一の場合には、データ転送の設定をバスリセット発生前の状態に戻す（ステップS73）。即ち、図20のステップS44で記憶したバスリセット発生時点までの転送済みデータサイズや、ステップS47で記憶したセグメントの内容やセグメント番号などに基づいて、図11のD5に示す位置（バスリセット発生時点の続き）からデータ転送を再開できるように、データ転送の設定をバスリセット前の状態に戻す。そして、継続フラグをオフに戻す（ステップS74）。この場合に、図17で説明したように、バスリセット前に既に転送を完了していたデータが消失しないように、ターゲットのデータバッファ上のデータをクリアしないようにする。

【0137】なお、ステップS74の後に、ステップS76のように転送データの先頭アドレスを記憶しておか

ないのは、バスリセット発生時点の続きからデータ転送を再開する場合には、バスリセット発生前に記憶しておいた先頭アドレスがそのまま使えるからである。

【0138】ステップS71でORBの内容がバスリセット前と同一でないと判断した場合、或いはステップS72で先頭アドレスがバスリセット前と同一でないと判断した場合には、データ転送の再開処理を行わず、継続フラグをオフに戻すと共に転送データの先頭アドレスを記憶しておく（ステップS75、S76）、即ち、この場合には、リードしたORBを全くの新規のORBとして処理することになる。

【0139】なお、ステップS75の場合には、リードしたORBを最初から処理することになるため、ステップS74と異なり、ターゲットのデータバッファのデータをクリアする。

【0140】6. 電子機器

次に、本実施形態のデータ転送制御装置を含む電子機器の例について説明する。

【0141】例えば図23(A)に電子機器の1つであるプリンタの内部ブロック図を示し、図24(A)にその外観図を示す。CPU（マイクロコンピュータ）510はシステム全体の制御などを行う。操作部511はプリンタをユーザが操作するためのものである。ROM516には、制御プログラム、フォントなどが格納され、RAM518はCPU510のワーク領域として機能する。表示パネル519はプリンタの動作状態をユーザに知らせるためのものである。

【0142】PHYデバイス502、データ転送制御装置500を介して、パーソナルコンピュータなどの相手ノードから送られてきた印字データは、バス504を介して印字処理部（プリンタエンジン）512に直接送られる。そして、印字データは、印字処理部512にて所与の処理が施され、プリントヘッダなどからなる印字部（データを出力するための装置）514により紙に印字されて出力される。

【0143】図23(B)に電子機器の1つであるスキヤナの内部ブロック図を示し、図24(B)にその外観図を示す。CPU520はシステム全体の制御などを行う。操作部521はスキヤナをユーザが操作するためのものである。ROM526には制御プログラムなどが格納され、RAM528はCPU520のワーク領域として機能する。

【0144】光源、光電変換器などからなる画像読み取り部（データを取り込むための装置）522により原稿の画像が読み取られ、読み取られた画像のデータは画像処理部（スキヤナエンジン）524により処理される。そして、処理後の画像データがバス505を介してデータ転送制御装置500に直接送られる。データ転送制御装置500は、この画像データにヘッダなどを付加することでパケットを生成し、PHYデバイス502を介し

てパーソナルコンピュータなどの相手ノードに送信する。

【0145】図23(C)に電子機器の1つであるCD-RWドライブの内部ブロック図を示し、図24(C)にその外観図を示す。CPU530はシステム全体の制御などを行う。操作部531はCD-RWをユーザが操作するためのものである。ROM536には制御プログラムなどが格納され、RAM538はCPU530のワーク領域として機能する。

【0146】レーザ、モータ、光学系などからなる読み取り&書き込み部（データを取り込むための装置又はデータを記憶するための装置）533によりCD-RW532から読み取られたデータは、信号処理部534に入力され、エラー訂正処理などの所与の信号処理が施される。そして、信号処理が施されたデータが、バス506を介してデータ転送制御装置500に直接送られる。データ転送制御装置500は、このデータにヘッダなどを付加することでパケットを生成し、PHYデバイス502を介してパーソナルコンピュータなどの相手ノードに送信する。

【0147】一方、PHYデバイス502、データ転送制御装置500を介して、相手ノードから送られてきたデータは、バス506を介して信号処理部534に直接送られる。そして、信号処理部534によりこのデータに所与の信号処理が施され、読み取り&書き込み部533によりCD-RW532に記憶される。

【0148】なお、図23(A)、(B)、(C)において、CPU510、520、530の他に、データ転送制御装置500でのデータ転送制御のためのCPUを別に設けるようにしてもよい。

【0149】また、図23(A)、(B)、(C)ではRAM501（データバッファに相当）がデータ転送制御装置500の外部に設けられているが、RAM501をデータ転送制御装置500に内蔵させてもよい。

【0150】本実施形態のデータ転送制御装置を電子機器に用いれば、バスに新たな電子機器が接続されてバスリセットが発生した場合にも、バスリセットを原因とする不具合の発生が防止される。これにより、電子機器の誤動作を防止できる。

【0151】また本実施形態のデータ転送制御装置を電子機器に用いれば、高速なデータ転送が可能になる。従って、ユーザがパーソナルコンピュータなどによりプリントアウトの指示を行った場合に、少ないタイムラグで印字が完了するようになる。また、スキヤナへの画像取り込みの指示の後に、少ないタイムラグで読み取り画像をユーザは見るようになる。また、CD-RWからのデータの読み取りや、CD-RWへのデータの書き込みを高速に行うことができるようになる。

【0152】また本実施形態のデータ転送制御装置を電子機器に用いることで、CPU上で動作するファームウ

エアの処理負担が軽減され、安価なCPUや低速のバスを用いることが可能になる。更に、データ転送制御装置の低コスト化、小規模化を図れるため、電子機器の低コスト化、小規模化も図れるようになる。

【0153】なお本実施形態のデータ転送制御装置を適用できる電子機器としては、上記以外にも例えば、種々の光ディスクドライブ（CD-ROM、DVD）、光磁気ディスクドライブ（MO）、ハードディスクドライブ、TV、VTR、ビデオカメラ、オーディオ機器、電話機、プロジェクタ、パーソナルコンピュータ、電子手帳、ワードプロセッサなど種々のものを考えることができる。

【0154】なお、本発明は本実施形態に限定されず、本発明の要旨の範囲内で種々の変形実施が可能である。

【0155】例えば、本発明のデータ転送制御装置の構成は、図7に示す構成が特に望ましいが、これに限定されるものではない。

【0156】また、データ転送期間中にバスリセットが発生したか否かの判断手法、コマンドの比較手法、データ転送の再開の手法も、本実施形態で説明した手法が特に望ましいが、これに限定されるものではない。

【0157】また、本発明はIEEE1394におけるバスリセットに特に有用だが、これ以外にも、少なくともノードのトポロジー情報をクリアするようなリセットであれば適用できる。

【0158】また、本発明は、IEEE1394規格でのデータ転送に適用されることが特に望ましいが、これに限定されるものではない。例えばIEEE1394と同様の思想に基づく規格やIEEE1394を発展させた規格におけるデータ転送にも本発明は適用できる。

【図面の簡単な説明】

【図1】IEEE1394の層構造について示す図である。

【図2】SBP-2について説明するための図である。

【図3】SBP-2のデータ転送処理の概略について説明するための図である。

【図4】データをインシエータからターゲットに転送する場合のコマンド処理について説明するための図である。

【図5】データをターゲットからインシエータに転送する場合のコマンド処理について説明するための図である。

【図6】図6（A）、（B）、（C）は、ページテーブルについて説明するための図である。

【図7】本実施形態のデータ転送制御装置の構成例を示す図である。

【図8】ターゲット側（ファームウェア）の処理の概要を示すフローチャートである。

【図9】インシエータ側（デバイスドライバ）の処理の概要を示すフローチャートである。

【図10】図10（A）、（B）、（C）は、二重印刷の問題について説明するための図である。

【図11】データ転送期間中にバスリセットが発生し、且つ、バスリセットの前後でORBの内容が同一の場合に、データ転送を継続して再開する手法について説明するための図である。

【図12】ORBの内容比較について説明するための図である。

【図13】継続フラグについて説明するための図である。

【図14】データ転送を再開するアドレスを特定するための情報について説明するための図である。

【図15】印刷コマンドを含む最初のコマンドブロックORBが来るまで、コマンド比較処理を繰り返す手法について説明するための図である。

【図16】ステータスのライト中にバスリセットが発生し、ACKミッシングとなった場合に、デッド状態に移行する手法について説明するための図である。

【図17】バスリセット時にインシエータに未だ転送していないデータを破棄せずに保持しておく手法について説明するための図である。

【図18】バスリセット発生時（リコネクト時）の本実施形態の詳細な処理例について示すフローチャートである。

【図19】バスリセット発生時（リコネクト時）の本実施形態の詳細な処理例について示すフローチャートである。

【図20】バスリセット発生時（リコネクト時）の本実施形態の詳細な処理例について示すフローチャートである。

【図21】通常時の本実施形態の詳細な処理例について示すフローチャートである。

【図22】通常時の本実施形態の詳細な処理例について示すフローチャートである。

【図23】図23（A）、（B）、（C）は、種々の電子機器の内部ブロック図の例である。

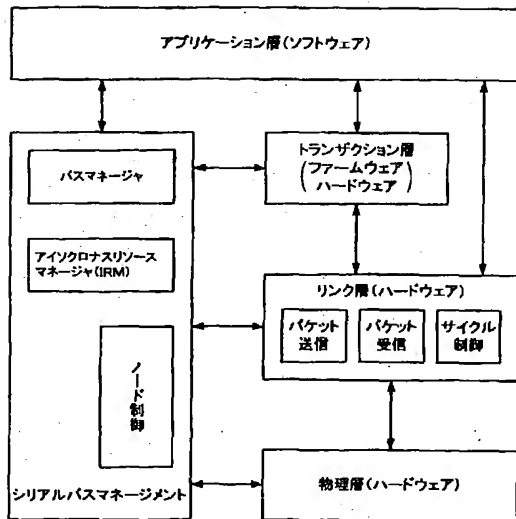
【図24】図24（A）、（B）、（C）は、種々の電子機器の外観図の例である。

【符号の説明】

- | | |
|----|-----------------|
| 10 | データ転送制御装置 |
| 12 | PHYデバイス |
| 14 | リンクデバイス |
| 16 | CPU |
| 18 | データバッファ |
| 20 | ファームウェア（F/W） |
| 30 | コミュニケーション部（COM） |
| 40 | マネージメント部（MNG） |
| 50 | プリントタスク部（PRT） |
| 60 | フェッチ部（FCH） |
| 62 | 判断部 |

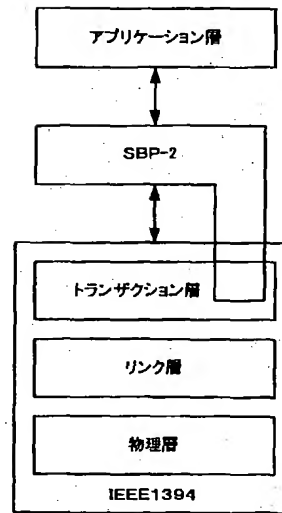
- 64 コマンド記憶部
- 66 コマンド比較部
- 68 アドレス記憶部
- 70 アドレス比較部
- 72 データ転送再開部

【図1】

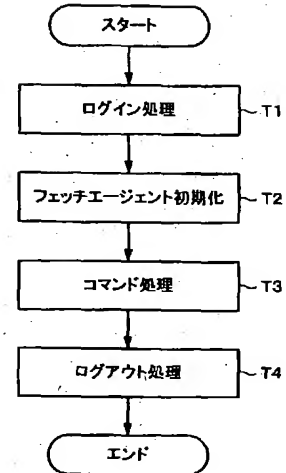


- 100 パーソナルコンピュータ
- 102 デバイスドライバ
- 104 データバッファ
- 110 情報記憶媒体

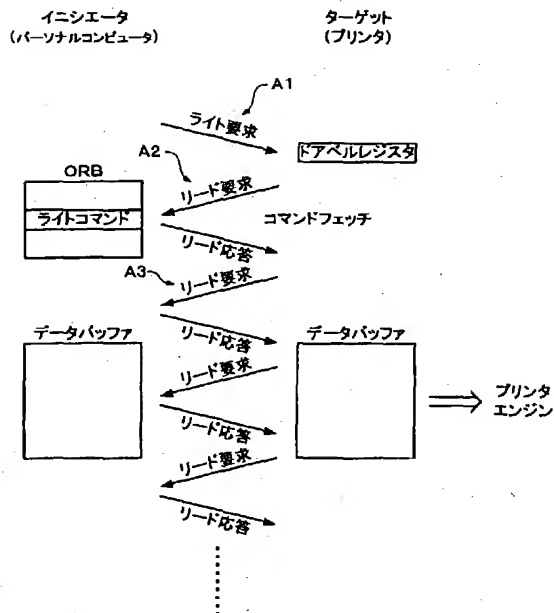
【図2】



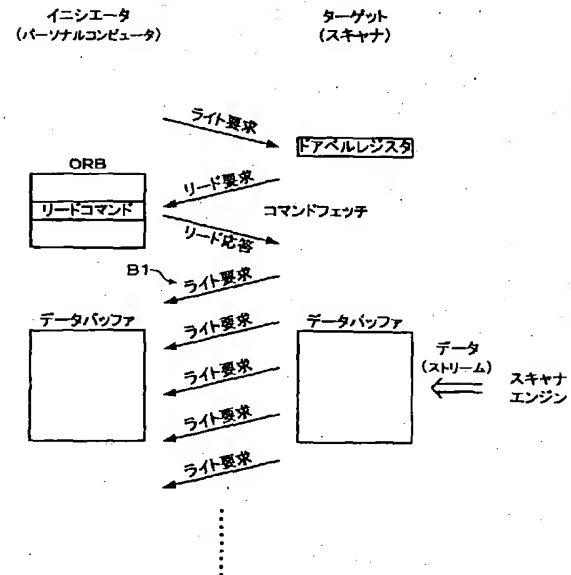
【図3】



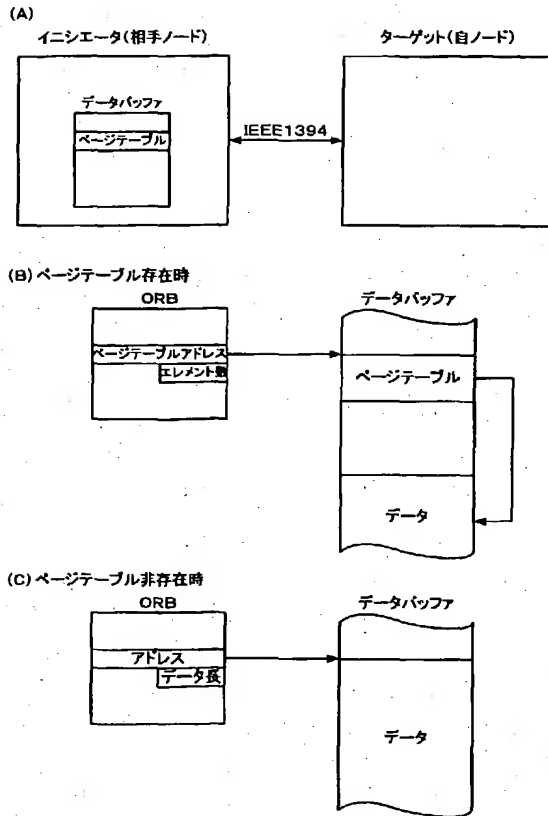
【図4】



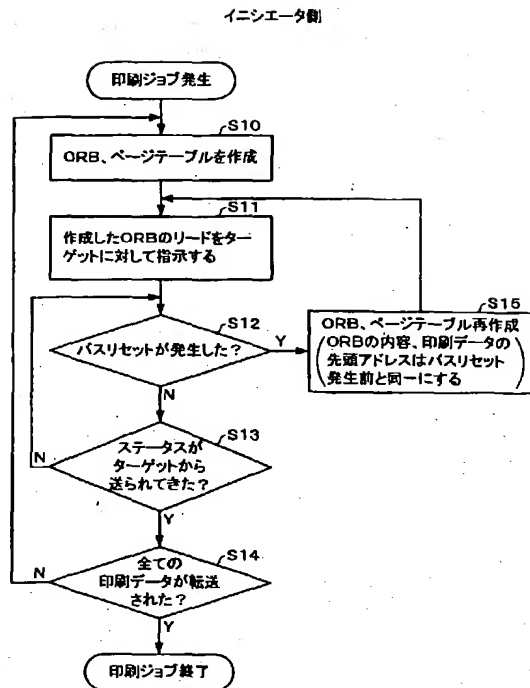
【図5】



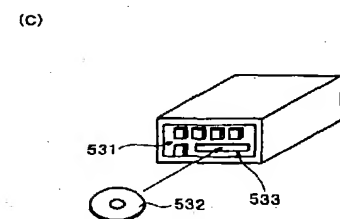
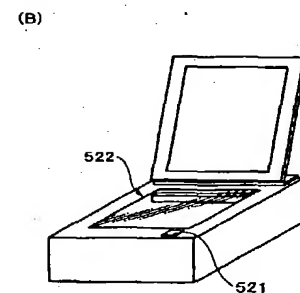
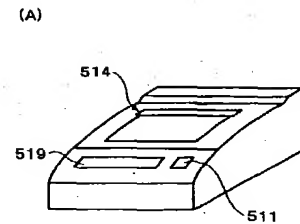
【図6】



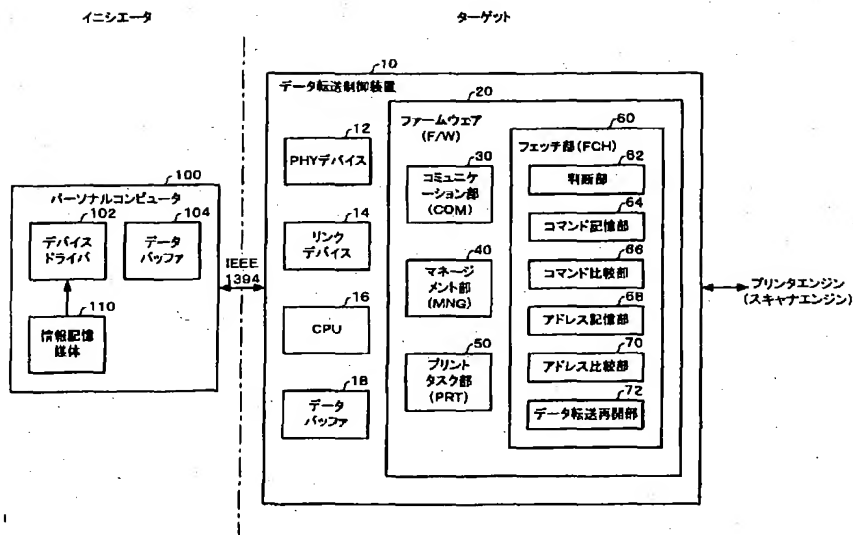
【図9】



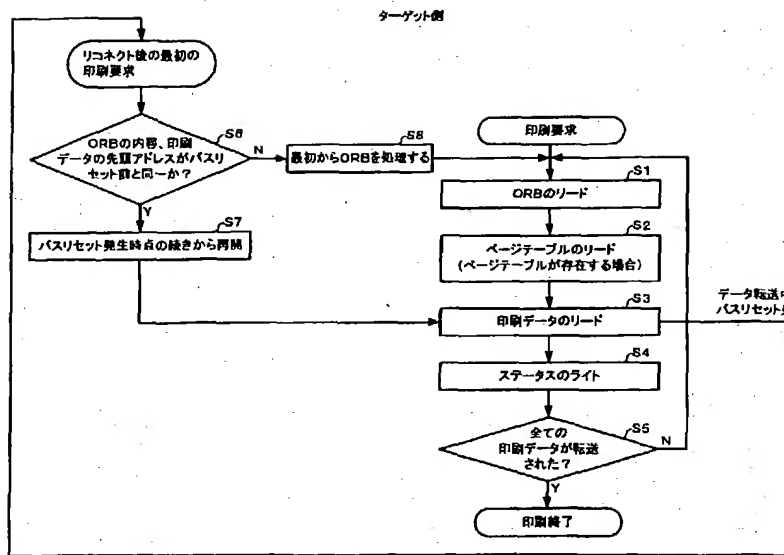
【図24】



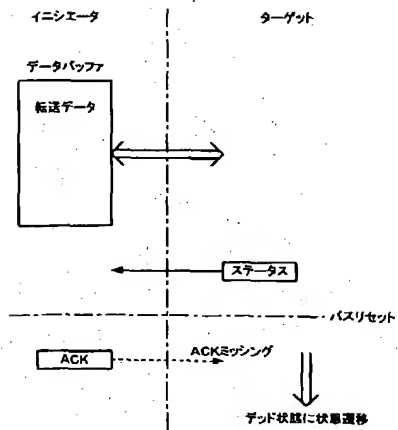
【図7】



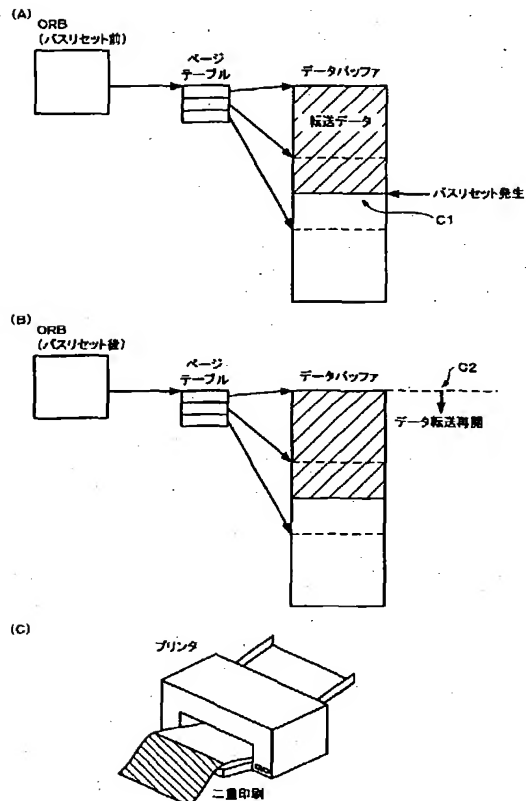
【図8】



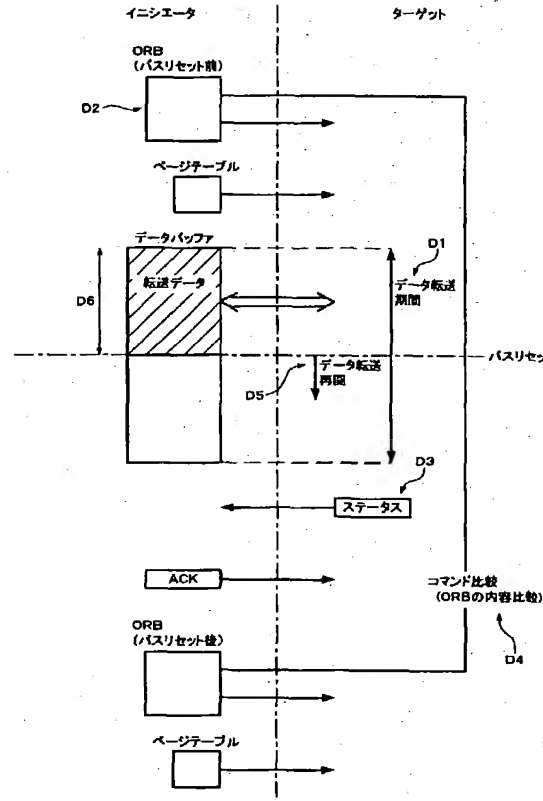
【図16】



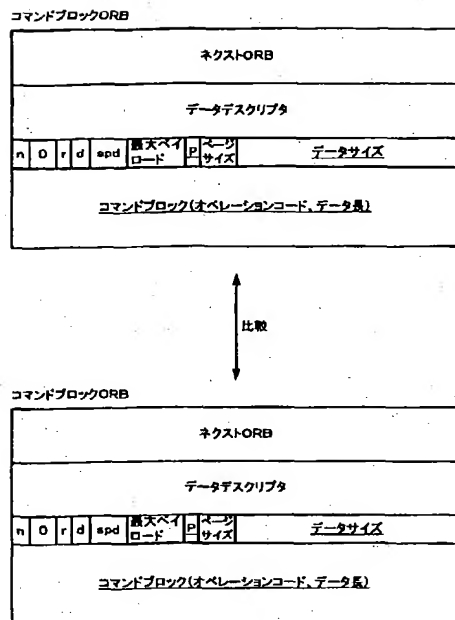
【図10】



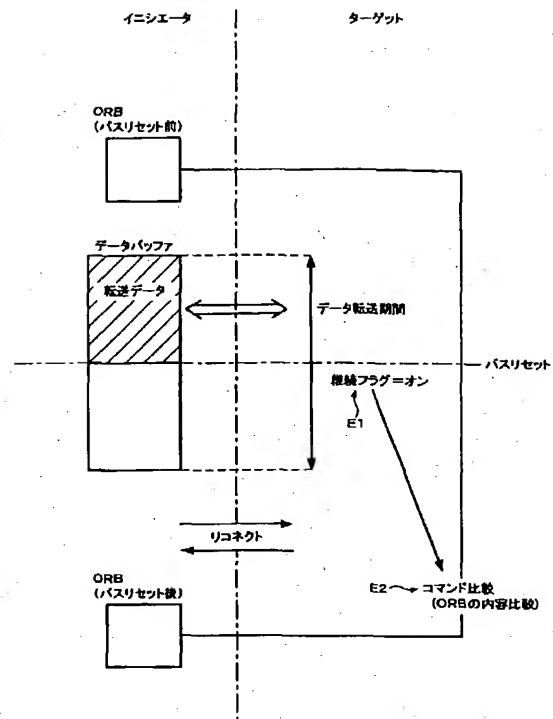
【図11】



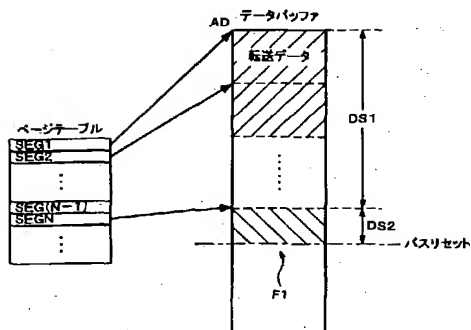
【図12】



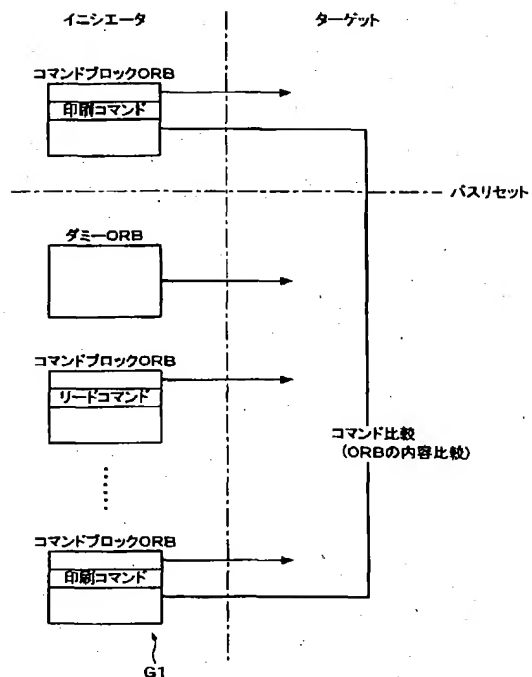
【図13】



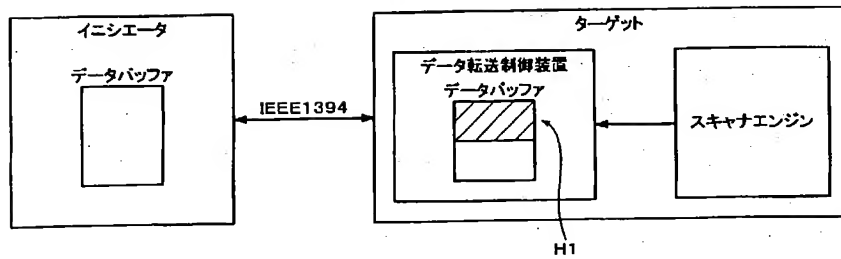
【図14】



【図15】

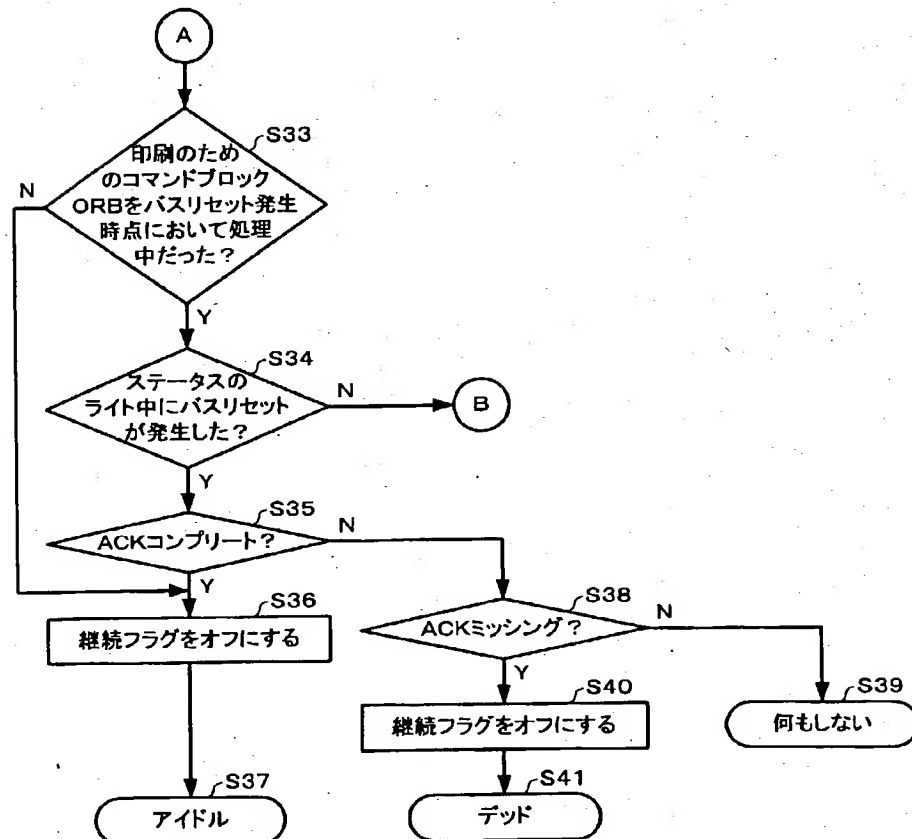


【図17】



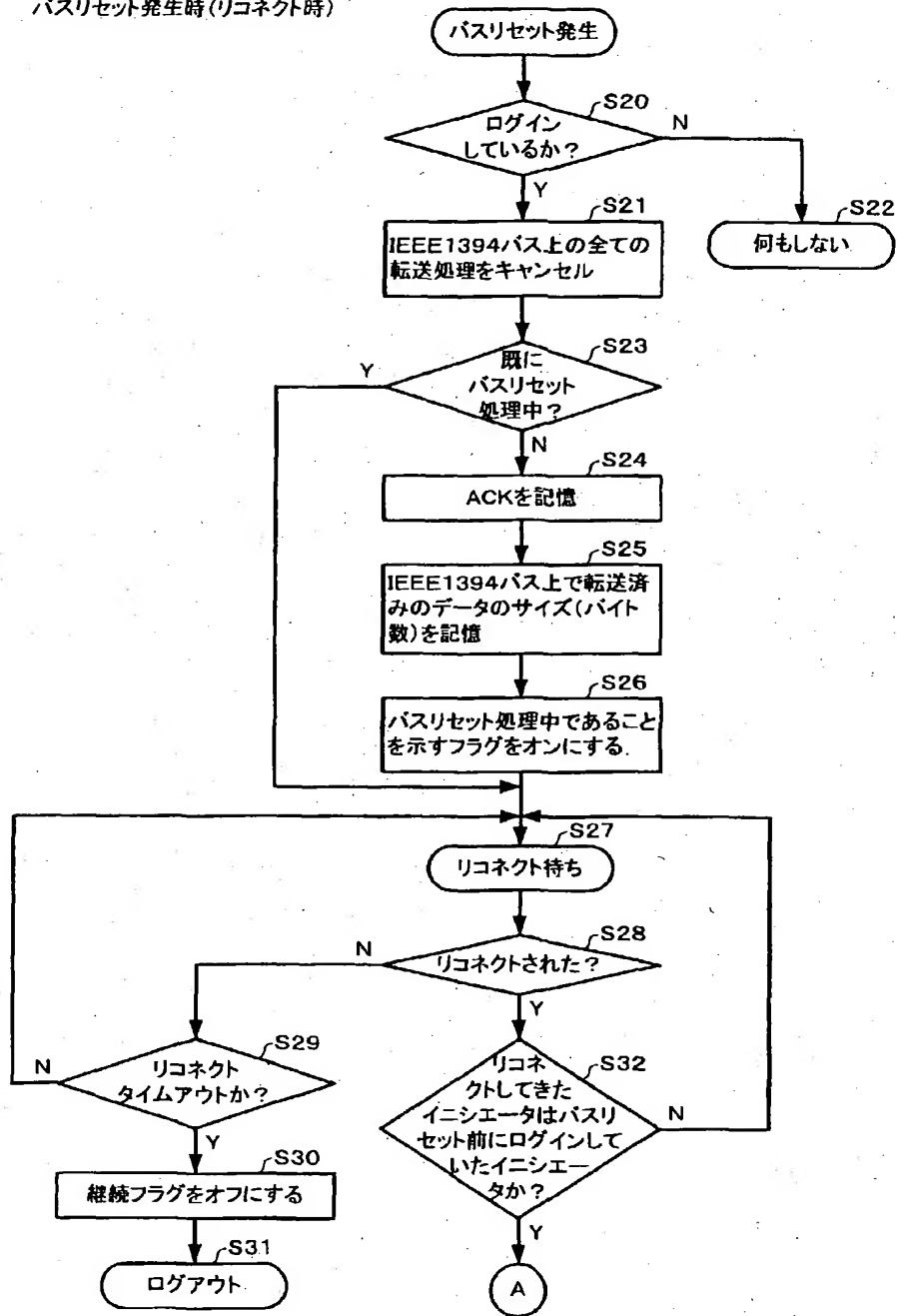
【図19】

バスリセット発生時(リコネクト時)



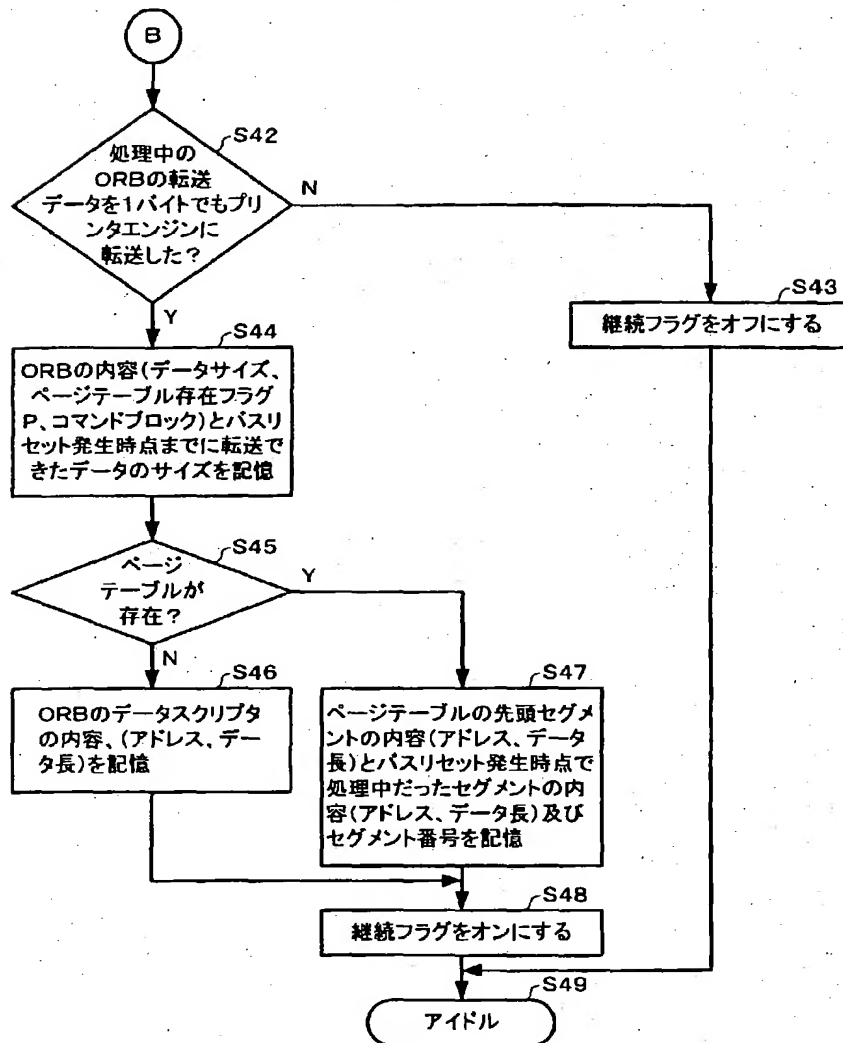
【図18】

バスリセット発生時(リコネクト時)

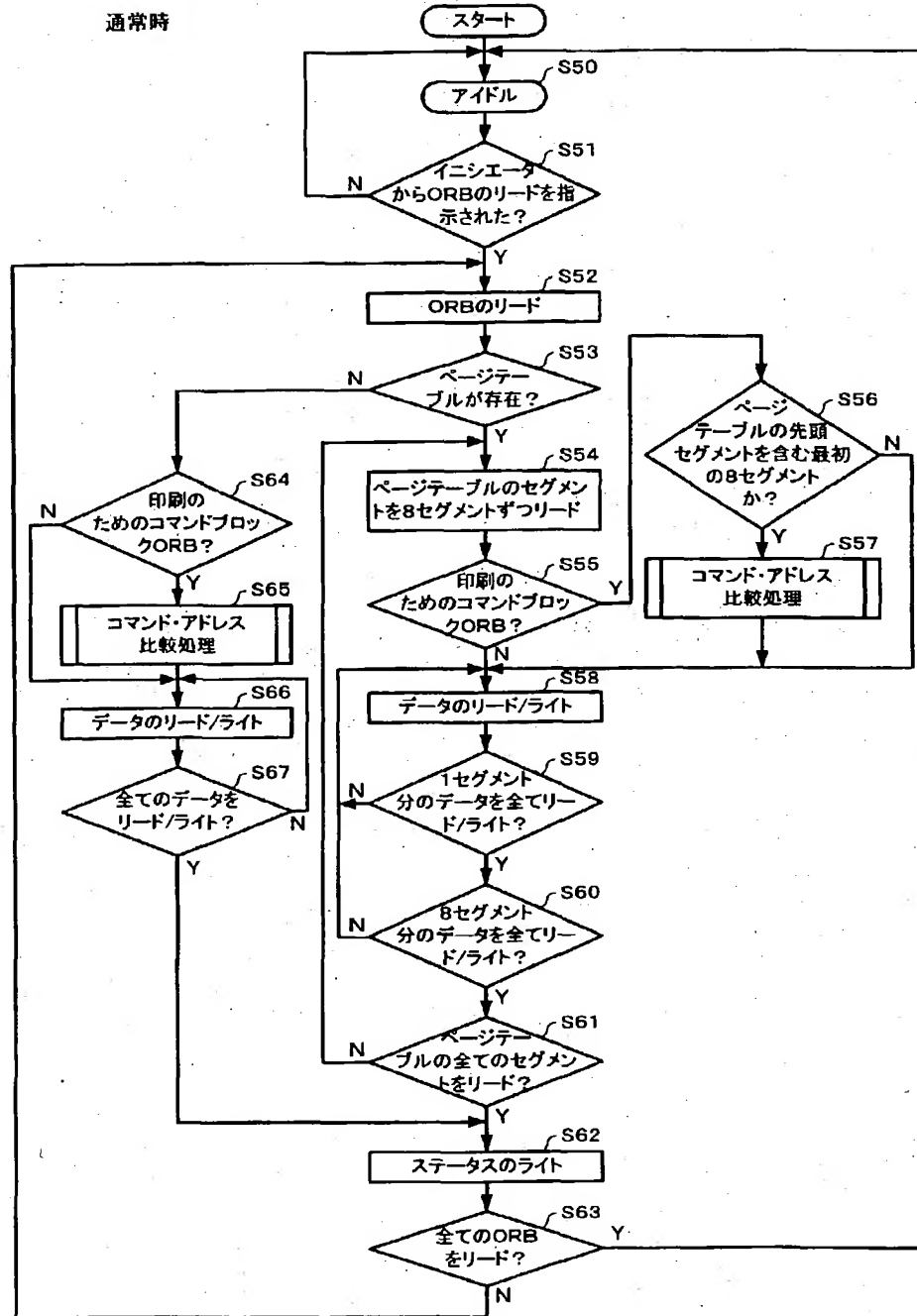


【図20】

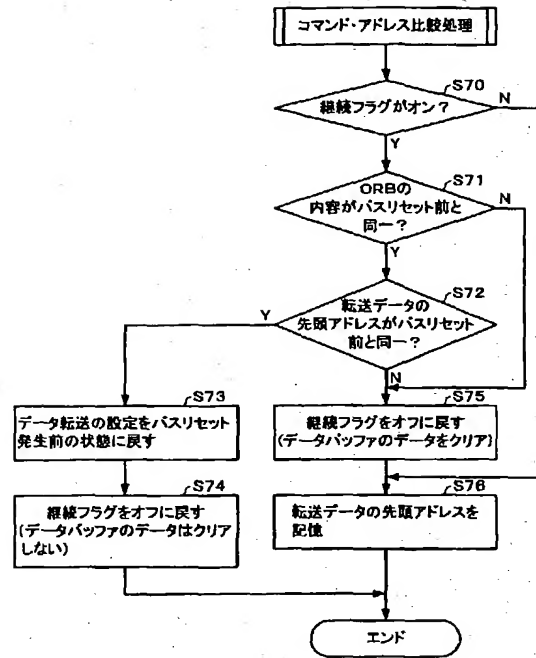
バスリセット発生時(リコネクト時)



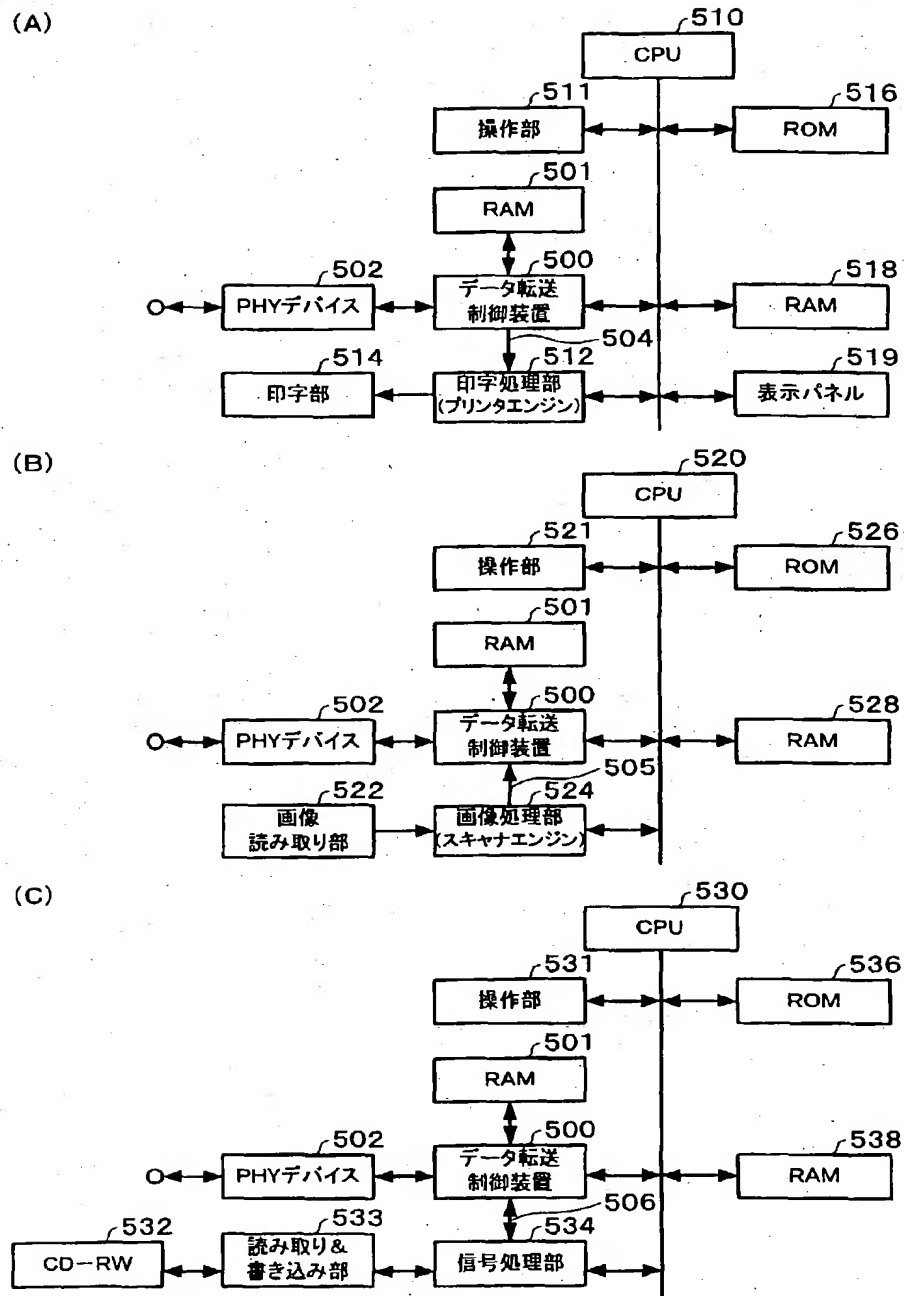
【図 21】



【図22】



【図 23】



フロントページの続き

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-177537

(43)Date of publication of application : 29.06.2001

(51)Int.Cl.

H04L 12/28

G06F 13/38

(21)Application number : 11-361103

(71)Applicant : SEIKO EPSON CORP

(22)Date of filing : 20.12.1999

(72)Inventor : MATSUNAGA KOSUKE

KANAI HIROYUKI

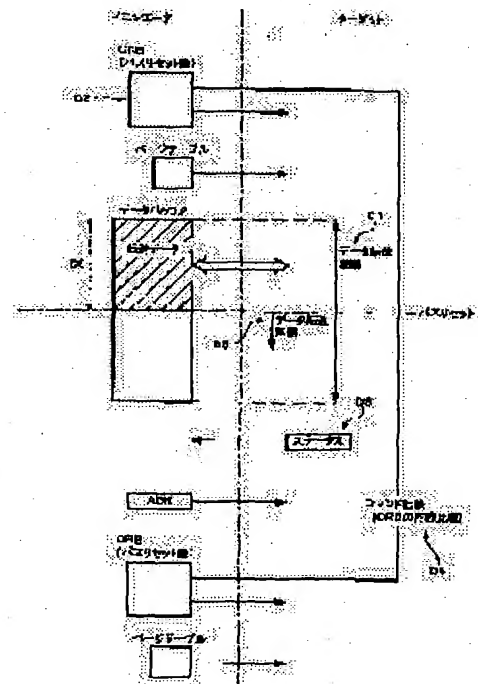
FUJITA SHINICHIRO

(54) DATA TRANSFER CONTROLLER, INFORMATION STORAGE MEDIUM AND ELECTRONIC EQUIPMENT

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a data transfer controller capable of eliminating deficiency generated at a time when a reset clearing the topology information of a node takes place, an information storage medium and electronic equipment.

SOLUTION: When a bus reset takes place during data transfer and also, the contents of an ORB before and after the bus reset are the same in the data transfer controller of an IEEE 1394 standard, data transfer is restarted by continuing from the point of time when the bus reset takes place to prevent a printer from performing double printing. When the bus reset takes place during data transfer, a continuity flag is made on. A command block ORB including a printing command, first transferred after the bus reset is adopted as a comparison object of the ORB before the bus reset. When an ACK is not returned from an initiator due to the bus reset, its state is shifted to a dead state. Transfer data that is not transferred to the initiator yet at the point of time when the bus reset occurs among transfer data from a scanner is not discarded but held.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision
of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] The data transfer control unit for the data transfer between two or more nodes connected to a bus characterized by providing the following. A judgment means to judge whether the reset which clears the topology information on a node during [after data transfer begins between partner nodes until it completes] the data transfer occurred. The contents of the 1st command packet for the data transfer operation demand transmitted from the partner node before generating of this reset. A command comparison means to compare the content of the 2nd command packet for the data transfer operation demand transmitted from the partner node after generating of this reset. A resumption means to resume data transfer from a continuation of the data transfer at the reset generating time when it is judged that the reset which clears the topology information on a node occurred during the aforementioned data transfer and it is judged that the above 1st and the 2nd command packet are the same contents.

[Claim 2] the generating time of the reset whose aforementioned judgment means clears the topology information on a node in a claim 1 -- setting -- the command packet of the above 1st for a data-transfer operation demand -- under processing -- it is -- and -- this -- the data-transfer control unit characterized by for this reset to judge that it generated during the aforementioned data transfer when data transfer by the 1st command packet has already been performed and has not transmitted the status of data-transfer completion to a partner node

[Claim 3] The data transfer control unit characterized by setting to ON the continuation flag which shows that data transfer continues and may be resumed when the reset whose aforementioned judgment means clears the topology information on a node judges that it generated during the data transfer in a claim 1 or 2.

[Claim 4] The data transfer control unit characterized by including a command storage means to memorize the information for specifying the address which resumes data transfer in a claim 1 or either of 3, and the content of the command packet of the above 1st for a data transfer operation demand by the time data transfer is resumed after generating of this reset.

[Claim 5] The data transfer control unit characterized by adopting the command packet for the data transfer operation demand first transmitted in the command packet transmitted from the partner node after generating of reset whose aforementioned command comparison means clears the topology information on a node in a claim 1 or either of 4 as a command packet of the above 2nd set as the comparison object of the command packet of the above 1st.

[Claim 6] The data transfer control unit characterized by changing a state transition into a data transfer improper state when generating of the reset which clears the topology information on a node becomes a factor and acknowledgement does not come on the contrary from a partner node in a claim 1 or either of 5, although the status of data transfer completion was transmitted to the partner node.

[Claim 7] It is the data transfer control unit characterized by holding about the transfer data which have not yet been transmitted to a partner node in the transfer data transmitted from the upper device in a claim 1 or either of 6 at the generating time of the reset which clears the topology information on a node, without canceling.

[Claim 8] The data transfer control unit characterized by the aforementioned reset being the bus reset defined in the specification of IEEE1394 in a claim 1 or either of 7.

[Claim 9] It is an information-storage medium including the program for controlling the data transfer between a claim 1 or one data transfer control unit of 8. When the reset which clears the topology information on a node occurs during the data transfer The information-storage medium characterized by including the program for creating the 2nd command packet of the same content as the 1st command packet *for the data transfer operation demand transmitted before generating of this reset, and carrying out a transfer request to a data transfer control unit.*

[Claim 10] Electronic equipment characterized by including the data transfer control unit of a claim 1 or either of 8, the equipment which performs given processing to the data received from the partner node through the aforementioned data transfer control unit and the bus, and the equipment for outputting or memorizing the data with which processing was performed.

[Claim 11] Electronic equipment characterized by including the data transfer control unit of a claim 1 or either of 8, the equipment which performs given processing to the data transmitted to a partner node through the aforementioned data transfer control unit and a bus, and the equipment for incorporating the data with which processing is performed.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[The technical field to which invention belongs] Especially this invention relates to the data transfer control unit, the information-storage medium, and electronic equipment for performing data transfer which applied to specification, such as IEEE1394, correspondingly among two or more nodes connected to a bus about a data transfer control unit, an information-storage medium, and electronic equipment.

[0002]

Background Art and Problem(s) to be Solved by the Invention] In recent years, the interface specification called IEEE1394 is in the limelight. This IEEE1394 standardizes the high-speed serial bus interface which can respond also to the multimedia of the next generation. According to this IEEE1394, the data with which real time nature, such as a dynamic image, is demanded can also be treated. Moreover, not only the peripheral device of computers, such as a printer, a scanner, a CD-RW drive, and a hard disk drive, but home electric appliances, such as a video camera, and VTR, TV, are connectable with the bus of IEEE1394. For this reason, it is expected as what can promote digitization of electronic equipment by leaps and bounds.

[0003] Now, in this IEEE1394, electronic equipment is newly connected to a bus, or electronic equipment is removed from a bus, and if the node connected to a bus fluctuates, the so-called bus reset will occur. And when bus reset occurs, the topology information on a node is cleared and it reconfigures topology information automatically after that. That is, tree discernment (determination of a root node) and self-discernment are performed after generating of bus reset, and management nodes, such as an isochronous resource manager, are determined after that. And the usual packet transfer is resumed.

[0004] Thus, in IEEE1394, since it reconfigures topology information automatically after bus reset, extraction and insertion (hot plug) of the so-called cable in a hot state is attained. For this reason, like the usual home electric appliances, such as VTR, a general user comes to be able to free extraction and insertion of the cable to electronic equipment, and can be useful to the so-called spread of home network systems.

[0005] However, in devices, such as a printer connected to the bus of IEEE1394, and a scanner, it became clear that generating of this bus reset becomes a factor and the following problems arise.

[0006] That is, if bus reset occurs during a transfer of print data on the bus of IEEE1394, initiators, such as a personal computer, will redo a transfer of print data again from the beginning. Therefore, to the printer which is a target, a part of print data will be sent to a duplex, and the problem of incorrect printings, such as double printing, arises.

[0007] Moreover, with a scanner, once a head begins to move, a head cannot be returned and the same data cannot be acquired again. Therefore, after generating of bus reset, even if an initiator tends to redo data transfer again from the beginning, there is a problem that data transfer is uncontinuable.

[0008] In addition, there are some which are indicated by JP,11-194902,A as conventional technology which solves the fault produced by generating of bus reset. With this conventional technology, if bus reset occurs, data processing is held, and after a network configuration is reconstructed, data processing

will be resumed.

[0009] However, with this conventional technology, the transfer data which only resend transfer data after bus reset generating, and are resent are not carrying out about judgment of being a continuation of the transfer data before bus reset generating. Therefore, depending on this conventional technology, the problem of double printing is unsolvable.

[0010] this invention is made in view of the above technical technical problems, and the place made into the purpose is to offer the data transfer control unit, the information-storage medium, and electronic equipment which can cancel the fault produced when the reset which clears the topology information on a node occurs.

[0011]

[Means for Solving the Problem] In order to solve the above-mentioned technical problem, this invention is a data transfer control unit for the data transfer between two or more nodes connected to a bus. A judgment means to judge whether the reset which clears the topology information on a node during [after data transfer begins between partner nodes until it completes] the data transfer occurred, The content of the 1st command packet for the data transfer operation demand transmitted from the partner node before generating of this reset, A command comparison means to compare the content of the 2nd command packet for the data transfer operation demand transmitted from the partner node after generating of this reset, When it is judged that the reset which clears the topology information on a node occurred during the aforementioned data transfer and it is judged that the above 1st and the 2nd command packet are the same contents It is characterized by including a resumption means to resume data transfer from a continuation of the data transfer at the reset generating time.

[0012] According to this invention, it is judged whether the reset which clears the topology information on a node occurred during the data transfer. Moreover, the content of the 1st command packet transmitted before reset generating is compared with the content of the 2nd command packet transmitted after reset generating. And if it is judged that reset occurred during the data transfer and it is judged that the 1st and 2nd command packet is the same content, data transfer will come to be resumed from the continuation at the reset generating time (from the next data of the data which are, for example at the reset generating time, and completed the transfer).

[0013] On the other hand, when it is judged that the 1st and 2nd command packet is not the same content, the 2nd command packet after reset generating comes to be processed from the beginning.

[0014] Therefore, according to this invention, when the partner node has carried out the transfer request of the command packet of the same content as reset generating before after reset generating, data transfer can be resumed from the continuation at the reset generating time. It follows, for example, data will overlap the device of the upper layer of a data transfer control unit, and will be transmitted to it, and the problem of the upper device malfunctioning can be solved.

[0015] in addition, the generating time of the reset whose aforementioned judgment means clears the topology information on a node -- setting -- the command packet of the above 1st for a data transfer operation demand -- under processing -- it is -- and -- this -- when data transfer by the 1st command packet has already been performed and has not transmitted the status of data transfer completion to a partner node, it is desirable for this reset to judge that it generated during the aforementioned data transfer

[0016] Moreover, this invention is characterized by setting to ON the continuation flag which shows that data transfer continues and may be resumed, when the reset to which the aforementioned judgment means clears the topology information on a node judges that it generated during the data transfer. If it does in this way, when a continuation flag will be OFF, it can avoid performing comparison processing of a command packet. Therefore, content comparison processing of an unnecessary command packet ceases to be performed, and mitigation-ization of a processing burden can be attained.

[0017] Moreover, this invention is characterized by including a command storage means to memorize by the time data transfer is resumed after generating of this reset of the information for specifying the address which resumes data transfer, and the content of the command packet of the above 1st for a data transfer operation demand. If such information is memorized, the restart procedure of data transfer can

be realized by simple processing.

[0018] Moreover, this invention is characterized by the aforementioned command comparison means adopting the command packet for the data transfer operation demand first transmitted in the command packet transmitted from the partner node after generating of the reset which clears the topology information on a node as a command packet of the above 2nd set as the comparison object of the command packet of the above 1st. If it does in this way, it will come to be transferred that content comparison processing of a command packet is performed until the command packet for the first data transfer operation demand is transmitted. It can prevent that content comparison processing of an unnecessary command packet is performed by this, and mitigation-ization of a processing burden can be attained.

[0019] Moreover, although this invention transmitted the status of data transfer completion to the partner node, when generating of the reset which clears the topology information on a node becomes a factor and acknowledgement does not come on the contrary from a partner node, it is characterized by changing a state transition into a data transfer improper state. Thus, when acknowledgement does not come on the contrary from a partner node, it becomes unknown whether the partner node received the status. Therefore, if data transfer is started from the continuation at the reset generating time in such a case, *mistaken data transfer may be performed*. Since a state transition is changed into a data transfer improper state when according to this invention the reset which clears the topology information on a node becomes a factor and acknowledgement does not come on the contrary from a partner node, the situation where data transfer which made the mistake in being such is performed can be prevented.

[0020] Moreover, this invention is characterized by holding without canceling about the transfer data which have not yet been transmitted to a partner node in the transfer data transmitted from the upper device at the generating time of the reset which clears the topology information on a node. If it does in this way, the data incorporated with the scanner etc. can prevent fault, such as being lost by generating of reset.

[0021] In addition, it is desirable that it is the bus reset by which the aforementioned reset is defined in the specification of IEEE1394 by this invention.

[0022] Moreover, this invention is an information-storage medium including the program for controlling the data transfer between one of the above-mentioned data transfer control units. When the reset which clears the topology information on a node occurs during the data transfer The 2nd command packet of the same content as the 1st command packet for the data transfer operation demand transmitted before generating of this reset is created, and it is characterized by including the program for carrying out a transfer request to a data transfer control unit. If it does in this way, the situation where the mistaken data transfer restart procedure will be performed can be prevented, and generating of the fault from which generating of reset becomes a factor can be prevented.

[0023] Moreover, the electronic equipment concerning this invention is characterized by including one of the above-mentioned data transfer control units, the equipment which performs given processing to the data received from the partner node through the aforementioned data transfer control unit and the bus, and the equipment for outputting or memorizing the data with which processing was performed. Moreover, the electronic equipment concerning this invention is characterized by including one of the above-mentioned data transfer control units, the equipment which performs given processing to the data transmitted to a partner node through the aforementioned data transfer control unit and a bus, and the equipment for incorporating the data with which processing is performed.

[0024] According to this invention, the situation which fault produces to a system by generating of the reset which clears the topology information on a node can be prevented, and it can prevent that electronic equipment malfunctions. Moreover, improvement in the speed of data transfer can be attained, and low-cost-izing of electronic equipment, improvement in the speed of processing of electronic equipment, etc. can be attained.

[0025]

[Embodiments of the Invention] Hereafter, the suitable operation gestalt of this invention is explained in detail using a drawing.

[0026] 1. IEEE1394 -- explain IEEE1394 briefly first

[0027] 1.1 In outline IEEE1394 (IEEE1394 -1995, P1394.a), the high-speed data transfer of 100 - 400Mbps is possible. Moreover, connecting to a bus the node from which a transfer rate differs is also allowed.

[0028] It connects in the shape of a tree, and connection of 63 nodes of each node is attained by the maximum into one bus. In addition, if a bus bridge is used, it is also possible to connect about 64000 nodes.

[0029] In IEEE1394, the asynchronous transmission and the isochronous transfer are prepared as the transmittal mode of a packet. An asynchronous transmission is the suitable transmittal mode for the data transfer as which reliability is required here, and an isochronous transfer is the suitable transmittal mode for data transfers as which real time nature is required, such as a dynamic image and voice.

[0030] 1.2 The layer structure (protocol composition) of layer-structure IEEE1394 is shown in drawing 1.

[0031] The protocol of IEEE1394 is constituted by a transaction layer, a link layer, and the physical layer. Moreover, it acts as the monitor of a transaction layer, a link layer, and the physical layer, or serial bus management controls them, and offers control of a node, and the various functions for resource management of a bus.

[0032] A transaction layer carries out transactions, such as a lead transaction, a light transaction, and a lock transaction, through the interface which provides a high order layer with the interface (service) of a transaction unit, and a lower layer link layer offers.

[0033] Here, in a lead transaction, data are transmitted to a demand node from a response node. On the other hand, in a light transaction, data are transmitted to a response node from a demand node.

Moreover, in a lock transaction, data are transmitted to a response node from a demand node, and a response node processes to the data and answers a demand node.

[0034] A link layer offers the cycle control for data framing for addressing, data check, and packet transmission and reception, and an isochronous transfer etc.

[0035] A physical layer offers conversion to the electrical signal of the logical symbol used by the link layer, mediation of a bus, and the physical interface of a bus.

[0036] 1.3 As shown in SBP-2, now drawing 2, the protocol called SBP-2 (Serial Bus Protocol-2) is proposed as a protocol of a high order including the function of a part of transaction layer of IEEE1394.

[0037] SBP-2 are proposed here in order to enable use of the commands set of SCSI on the protocol of IEEE1394. If these SBP-2 are used, the minimum change is added to the commands set of SCSI currently used by the electronic equipment of the existing SCSI specification, and it can be used for the electronic equipment of IEEE1394 specification. Therefore, -izing of a design and development of electronic equipment can be carried out [easy]. Moreover, since not only the command of SCSI but a command peculiar to a device can be encapsulated and used, versatility is very high.

[0038] As shown in drawing 3, in SBP-2, login processing is first performed using the login ORB (Operation Request Block) created by the initiator (for example, personal computer) (Step T1). Next, a fetch agent's initialization is performed using Dummy ORB (Step T2). And command processing is performed using the command block ORB (normal command ORB) (Step T3), finally, Logout ORB is used and logout processing is performed (Step T4).

[0039] Here, in command processing of Step T3, as shown in A1 of drawing 4, an initiator transmits a light demand packet (publishing a light demand transaction), and carries out the ring of the doorbell register of a target. Then, as shown in A2, a target transmits a lead demand packet and the lead response packet to which an initiator corresponds is returned. Thereby, ORB (command block ORB) which the initiator created is fetched to the data buffer of a target. And a target analyzes the command contained in fetched ORB.

[0040] And when the command contained in ORB is a light command of SCSI, as shown in A3, a target transmits a lead demand packet to an initiator, and the lead response packet to which an initiator corresponds is returned. Thereby, the data stored in the data buffer of an initiator are transmitted to a target. And when a target is a printer, for example, the transmitted data are printed with a printer engine.

[0041] On the other hand, when the command contained in ORB is a lead command of SCSI, as shown in B1 of drawing 5, a target transmits a series of light demand packets to an initiator. By this, when a target is a scanner, the scanning data acquired with the scanner engine will be transmitted to the data buffer of an initiator.

[0042] According to these SBP-2, a target transmits a demand packet, when convenience of self is good (publishing a transaction), and it can transmit and receive data. Therefore, since it becomes unnecessary for an initiator and a target to synchronize and move, data transmission efficiency can be raised.

[0043] In addition, as a high order protocol of IEEE1394, the protocol called FCP (Function Control Protocol) besides SBP-2 is proposed.

[0044] Now, when performing data transfer between a target and an initiator, a page table may exist like drawing 6 (A) at the data buffer (storage means) of an initiator (partner node), or it may not exist.

[0045] And when a page table exists, as shown in drawing 6 (B), in ORB which the initiator created, the address and the number of elements of the page table are contained. And indirect addressing of the address (read-out address, write-in address) of transfer data is carried out using this page table.

[0046] On the other hand, when a page table does not exist, as shown in drawing 6 (C), the address and a data length are contained in ORB and direct addressing of the address of transfer data is carried out.

[0047] 1.4 In bus reset IEEE1394, if a power supply is switched on or extraction and insertion of a device occurs on the way, bus reset will occur. That is, each node is supervising voltage change of a port. And if a new node is connected to a bus and change arises on the voltage of a port, the node which detected this change will tell that bus reset occurred to other nodes on a bus. Moreover, the physical layer of each node tells that bus reset occurred to a link layer.

[0048] And if bus reset occurs in this way, topology information, such as Node ID, will be cleared. And it reconfigures topology information automatically after that. That is, tree discernment and self-discernment are performed after bus reset. Then, management nodes, such as an isochronous resource manager and cycle master and a bus manager, are determined. And the usual packet transfer is resumed.

[0049] Thus, the cable of electronic equipment can be freely taken out and inserted now, and in IEEE1394, since it reconfigures topology information automatically after bus reset, the so-called hot plug can be realized.

[0050] In addition, the transaction is canceled when bus reset occurs in the middle of a transaction. And the demand node which published the canceled transaction transmits a demand packet again, after reconfiguring topology information. Moreover, a response node must not return the response packet of the transaction canceled by bus reset to a demand node.

[0051] 2. Explain whole composition, next the whole data transfer control unit example of composition of this operation gestalt using drawing 7. In addition, although the case where the target which performs data transfer is a printer is taken and explained to an example between initiators below, this invention is not limited to this.

[0052] The data transfer control unit 10 of this operation gestalt contains the PHY device 12 (device of a physical layer), the link device 14 (device of a link layer), CPU16 (processor), a data buffer 18 (storage means), and a firmware 20 (processing means). In addition, the PHY device 12, the link device 14, CPU16, and data buffers 18 are arbitrary components, and the data transfer control unit 10 of this operation gestalt does not need to contain all of these components.

[0053] The PHY device 12 is a circuit for hardware realizing the protocol of the physical layer of drawing 1, and has the function to change into an electrical signal the logical symbol used by the link device 14.

[0054] The link device 14 is a circuit for hardware realizing a part of protocol of the link layer of drawing 1, and protocol of a transaction layer, and offers the various services for the packet transfer between nodes.

[0055] CPU16 performs control of the whole equipment and control of data transfer.

[0056] A data buffer 18 is a buffer which stores transfer data (packet) temporarily, and is constituted by hardware, such as SRAM, SDRAM, or DRAM. In addition, with this operation gestalt, a data buffer 18 functions as a packet storage means in which random access is possible.

[0057] A firmware 20 is a program containing the various manipulation routines (processing module) which operate on CPU16, and the protocol of a transaction layer is realized by the CPU16 grade which are this firmware 20 and hardware.

[0058] In addition, the device driver 102 which the personal computer 100 which is an initiator contains is a program containing the various manipulation routines for carrying out supervisory control of the peripheral device. This program is installed in a personal computer 100 using the information-storage medium 110 (FD, CD-ROM, DVD, ROM).

[0059] The program of a device driver 102 is downloaded through networks, such as the Internet, from the information-storage media (a hard disk, magnetic tape, etc.) which a host system has, and you may make it install it in a personal computer 100 here. Use of the information-storage medium which such a host system has is also included within the limits of this invention.

[0060] A firmware 20 (F/W) contains the communication section 30 (COM), the management section 40 (MNG), the print task section 50 (PRT), and the fetch section 60 (FCH).

[0061] Here, the communication section 30 is a processing module which functions as an interface between hardware, such as the link device 14.

[0062] The management section 40 (management agent) is a processing module which manages login, reconnection, logout, reset, etc. For example, when an initiator requires login of a target, this management section 40 will receive this login demand first.

[0063] The print task section 50 is a processing module which performs data transfer processing between the printer engines which are latter application layers (upper layer).

[0064] The fetch section 60 (a fetch agent, command block agent) is a processing module for executing the command which the command block ORB contains. Unlike the management section 40 which can treat only a single demand, the fetch section 60 can also treat the linked list of ORB which self fetched by the demand from an initiator.

[0065] The fetch section 60 contains the judgment section 62, the command storage section 64, the command comparator 66, the address storage section 68, the address comparator 70, and the data transfer resumption section 72.

[0066] The judgment section 62 performs processing which judges whether bus reset (reset which clears the topology information on a node in a wide sense) occurred during [which transmits print data between initiators (partner node)] the data transfer here.

[0067] The command storage section 64 is ORB (command block ORB.) transmitted from the initiator before generating of bus reset. Processing for memorizing the content of the command packet for a data transfer operation demand, when the time of bus reset occurring and reconnection are successful is performed in a wide sense.

[0068] The command comparator 66 performs processing which compares the content (content memorized by the command storage section 64) of ORB (command block ORB) transmitted from the initiator before generating of bus reset with the content of ORB transmitted from the initiator after generating of bus reset.

[0069] The address storage section 68 performs processing for memorizing the start address (the 1st address) of the transfer data (print data) transmitted between initiators.

[0070] The address comparator 70 performs processing which compares with the start address (the 2nd address) of the transfer data after bus reset generating the start address (the 1st address) memorized by the address storage section 68, when bus reset occurs.

[0071] When it is judged that bus reset generated the data transfer resumption section 72 during the data transfer and it is judged that the content of ORB (command block ORB) was in agreement, processing which resumes data transfer from a continuation (the next data of the data which it is at the bus reset generating time, and were transmitted) of the data transfer at the bus reset generating time is performed.

[0072] 3. Explain the outline of processing, next the outline of processing of this operation gestalt.

[0073] Drawing 8 is a flow chart which shows the outline of processing by the side of a target (firmware).

[0074] If there is a printing demand from an initiator, a target leads ORB from the data buffer of an

initiator (Step S1). And when a page table exists, based on the page table address (refer to drawing 6 (B)) included in ORB, a page table is led from the data buffer of an initiator (Step S2). Next, based on the led page table, print data are led from the data buffer of an initiator (Step S3). And if all the print data specified by the page table are led, the light of the status will be carried out and the statuses -- whether data transfer was successful -- will be told to an initiator (step S4). It is repeated until the above processing is transmitted to all print data (Step S5).

[0075] And with this operation gestalt, if bus reset occurs during a transfer of print data (data transfer period), the following processings will be performed to the printing demand of the beginning after reconnection.

[0076] That is, it first judges whether the content of ORB before bus reset, the start address of print data, and the content of ORB after bus reset and the start address of print data are the same (Step S6). And when it is judged that it is the same, data transfer is resumed from the continuation at the bus reset generating time (Step S7). On the other hand, when it is judged that it is not the same, it processes from the beginning by setting ORB after bus reset to new ORB (Step S8).

[0077] Drawing 9 is a flow chart which shows the outline of processing by the side of an initiator (device driver).

[0078] If the printing job from an application program occurs, an initiator will create ORB and the page table for printing, and will write them in a data buffer (Step S10). Next, it directs to lead created ORB to a target (Step S11.). A1 reference of drawing 4.

[0079] Next, it judges whether bus reset occurred (Step S12), and when it does not generate, it judges whether the status has been sent from the target (Step S13). And it judges whether when sent, all print data were transmitted (Step S14), when not transmitted, it returns to Step S10, and a printing job is ended when transmitted.

[0080] And with this operation gestalt, if it is judged that bus reset occurred at Step S12, an initiator re-creates ORB and a page table (Step S15), and directs to lead re-created ORB to a target (Step S11). In this case, an initiator (device driver) re-creates ORB so that the content of ORB before bus reset generating, the start address of print data, and the content of ORB after bus reset generating and the start address of print data may become the same.

[0081] 4. When bus reset occurred during the feature of this operation gestalt, now the transfer of print data, it became clear that the following problems arise.

[0082] For example, as shown in drawing 10 (A), suppose that bus reset occurred in the place to which data were transmitted to the position (address) shown in C1. In this case, all the transactions that it is at the bus reset generating time, and were under processing are canceled. Therefore, as shown in drawing 10 (B), the initiator which was demanding the transfer of print data before bus reset creates ORB for printing again after bus reset, and directs to redo a transfer of print data from the beginning at a target. For this reason, data transfer will be resumed from the position shown in C2 of drawing 10 (B), and a part of print data will be sent to a duplex. Consequently, the problem of double printing as shown in drawing 10 (C) occurs.

[0083] In order to solve such a problem, technique which is explained below is adopted with this operation gestalt.

[0084] That is, with this operation gestalt, first, as shown in D1 of drawing 11, it judges whether bus reset occurred during the data transfer (period after a transfer of print data begins between initiators until it completes). ORB (command block ORB) shown in D2 at the generating time of bus reset is more specifically under processing, and data transfer by the ORB has already been performed, and when the status of the data transfer completion shown in D3 is not transmitted to an initiator, bus reset judges that it generated during the data transfer.

[0085] Moreover, with this operation gestalt, as shown in D4 of drawing 11, an initiator succeeds in reconnection after generating of bus reset, and when the transfer request of new ORB has been created and carried out, the content of ORB before bus reset (1st command packet) is compared with the content of ORB after bus reset (2nd command packet).

[0086] And when it is judged that bus reset occurred during the data transfer and the content of ORB

before bus reset and the content of ORB after bus reset are judged to be the same contents, as shown in D5, data transfer is resumed from a continuation of the data transfer at the bus reset generating time (step S7 reference of drawing 8). That is, data transfer is resumed from the next data of the data which it is at the bus reset generating time, and had already completed the transfer.

[0087] On the other hand, when the content of ORB is not the same, ORB after bus reset is processed from the beginning as a new thing the case where bus reset does not occur during the data transfer, in front of bus reset, and in the back (step S8 reference of drawing 8).

[0088] Unlike the case of drawing 10 (B), by doing in this way, the double transfer of the transfer data of the portion shown in D6 of drawing 11 ceases to be carried out. Incorrect printing as shown in drawing 10 (C) stops therefore, arising. Moreover, since a double transfer is avoidable, the transfer time can also be shortened.

[0089] For example, since the double transfer of the transfer data is not carried out even if it does not resume data transfer from the continuation at the bus reset generating time like D5 when bus reset occurs in periods other than a data transfer period, it is satisfactory. Moreover, when bus reset occurs in periods other than a data transfer period, processing also becomes simple and a processing burden also becomes [the direction which processed ORB from the beginning] light rather than performing the restart procedure of data transfer.

[0090] Moreover, the technique of resuming data transfer from the continuation at the bus reset generating time is also always considered as different technique from this operation gestalt, without performing comparison processing of the content of ORB.

[0091] However, according to this technique, an initiator cancels transfer processing of print data after bus reset, and when ORB which is completely different bus reset before is created, the fault that data transfer will be resumed from D5 of drawing 11 arises, for example.

[0092] On the other hand, with this operation gestalt, although data transfer resumes from D5 of drawing 11 in being the same before and after the content of ORB resets [bus], since it is completely processed as new ORB when not the same, the above faults do not arise.

[0093] In addition, this operation gestalt is comparing various information in the case of content comparison of ORB. For example, as shown in drawing 12 , this operation gestalt is comparing the page table existence flag P and data size which the command block ORB contains, the operation code (code which distinguishes a printing command, a lead command, etc.) in the command block (commands set) field, and the data length. Moreover, when ORB contains the identification information (for example, sequence number) for discriminating ORB, this identification information is also compared. By comparing such information, it can judge now certainly whether ORB before and behind bus reset is the same by simple processing.

[0094] Now, as shown in E1 of drawing 13 , when bus reset judges that it generated during the data transfer, it is made to set to ON the continuation flag which shows that data transfer continues and may be resumed with this operation gestalt. For example, after generating of bus reset, when reconnection is successful, it is judged whether bus reset occurred during the data transfer and it is judged that it generated, a continuation flag is set to ON.

[0095] And as shown in E2 of drawing 13 , content comparison of ORB is performed on condition that this continuation bit is ON. That is, when a continuation flag is OFF, content comparison of ORB is not performed. It comes to be prevented that content comparison processing of unnecessary ORB is performed by this, and it can carry out [****]-izing of the processing burden of a firmware.

[0096] Moreover, with this operation gestalt, the information for specifying the address (D5 of drawing 11) which resumes data transfer, the content of ORB before bus reset, etc. are memorized, when a bus reset generating time and reconnection are successful (until data transfer is resumed after generating of bus reset by the wide sense). By memorizing such information, when ORB has been again transmitted from the initiator after bus reset, it can judge easily whether data transfer is resumed from the continuation at the bus reset generating time. Moreover, the restart procedure of data transfer can also be simplified now.

[0097] In addition, as information for specifying the address which resumes data transfer, the following

information can specifically be considered.

[0098] For example, as shown in drawing 14, suppose that the transfer data which the Nth segment SEGN of a page table specifies at the bus reset generating time were transmitted. In this case, as information for specifying the address (address shown in F1) which resumes data transfer The address (start address of transfer data) stored in the head segment SEG1 of a page table, The size DS 1 of data and the segment number of Segment N which were transmitted by Segments SEG1-SEG (N-1), the size DS 2 of the data transmitted by Segment N by the bus reset generating point in time, etc. can be considered.

[0099] Moreover, it is made to adopt ORB (command block ORB) which contains the printing command transmitted first in ORB transmitted from the initiator after bus reset generating as ORB set as the comparison object of ORB before bus reset with this operation gestalt.

[0100] For example, as shown in drawing 15, the command block ORB which contains a printing command after bus reset is not necessarily transmitted immediately. That is, before an initiator carries out the transfer request of the command block ORB containing a printing command, the transfer request of Dummy ORB and the management ORB may be carried out. Or in order to investigate the status of a printer etc., the transfer request of the command block ORB containing a lead command may be carried out.

[0101] With this operation gestalt, as shown in G1 of drawing 15, even if Dummy ORB etc. is transmitted, content comparison processing of ORB is not performed, but when the command block ORB containing a printing command has been transmitted, content comparison processing of ORB is performed for the first time. That is, execution of content comparison processing of ORB is transferred until the first command block ORB containing a printing command comes (it is made passing <a thing> on). It is prevented that content comparison processing of unnecessary ORB is performed by this, and it can carry out [****]-izing of the processing burden of a firmware.

[0102] Now, the generating time of bus reset is entire arbitration. therefore -- for example, although a target transmits the status of data transfer completion to an initiator as shown in drawing 16, generating of bus reset becomes a factor, and ACK (ACK -- complete) may not come on the contrary from an initiator, but it may become ACK missing

[0103] In such a case, although generating of bus reset became a factor, and an initiator could not receive the status, but the 1st case where it became ACK missing, and the initiator received the status and ACK was returned, the 2nd case where generating of bus reset became a factor and became ACK missing can be considered.

[0104] And in the 1st case of the above, it thinks that data transfer of the initiator was unsuccessful, and 1st processing in which the same ORB is again created after bus reset is performed. On the other hand, in the 2nd case of the above, an initiator thinks that data transfer was successful and performs 2nd processing in which following ORB is created after bus reset.

[0105] However, since only the information that it was ACK missing gets across to a target, a target cannot know any the initiator should perform between the above 1st and the 2nd processing. Therefore, if data transfer is started from the continuation at the bus reset generating time in such a case, mistaken data transfer may be performed.

[0106] Then, as shown in drawing 16, when generating of bus reset becomes a factor and acknowledgement does not come by this operation form on the contrary from an initiator, with it, it changes in the dead state (data transfer improper state). The situation where mistaken data transfer is performed can be prevented by doing in this way.

[0107] Now, although the case where a target was a printer was mainly explained above, when a target is a scanner, there are the following problems.

[0108] That is, with a scanner, once a head begins to move, there is a problem that a head cannot be returned and the same data cannot be acquired again (scan).

[0109] Then, it is made to hold with this operation form, about the transfer data which have not yet been transmitted to an initiator in the transfer data from a scanner engine (a wide sense the upper device) at the generating time of bus reset, without canceling, as shown in drawing 17. That is, although it

incorporated from the scanner engine and stored in the data buffer as shown in H1 of drawing 17 , it holds about the data which have not yet been transmitted to an initiator through the bus of IEEE1394, without canceling it, as a data buffer is not cleared.

[0110] When doing in this way and data transfer is resumed from the continuation at the bus reset generating time by the technique shown in drawing 11 after bus reset generating, fault -- the image data incorporated before bus reset generating will be lost -- can be prevented.

[0111] 5. Explain the detailed example of processing, next the detailed example of processing of this operation form using the flow chart of drawing 18 - drawing 22 .

[0112] Drawing 18 - drawing 20 are flow charts which show the detailed example of the processing at the time of bus reset generating (at the time of reconnection).

[0113] If bus reset occurs, a target judges first whether the initiator logs in (Step S20), and when logged in, it will cancel all transfer processings on the bus of IEEE1394 (transaction) (Step S21). On the other hand, when not logged in, even if bus reset occurs, since it is unnecessary, no special processings are carried out (Step S22).

[0114] Next, it judges whether bus reset processing is already started (Step S23). Thereby, when bus reset carries out multiple-times generating, the situation where the bus reset processing corresponding to it is repeated two or more times unnecessarily can be prevented.

[0115] Next, the state of ACK at the bus reset generating time (acknowledgement) is memorized (Step S24). Thereby, the situation where the contents of ACK immediately after bus reset will be erased can be prevented by the transaction (for example, transaction of reconnection) generated after that.

[0116] Next, the size (byte count) of data [finishing / a transfer] is memorized on the bus of IEEE1394 (Step S25). That is, the size (DS2 of drawing 14) of data [finishing / the transfer by the segment which was under processing at the bus reset generating time] is memorized. And the flag which shows that it is [bus reset] under processing because of judgment of Step S23 is turned ON (Step S26). That is, after that, if this flag is turned on [it], even if bus reset occurs, processing of Steps S24-S26 will not be performed.

[0117] Next, it becomes the reconnection waiting from an initiator (Step S27), and judges whether it was reconnected by the initiator (Step S28). And when not reconnected, it judges whether the reconnection time-out time specified in the reconnection field of Login ORB passed (Step S29). And when it passes, the continuation flag (flag which shows that data transfer continues and may be resumed) explained by drawing 13 is turned OFF (Step S30), and a state transition is changed into a logout state (Step S31).

[0118] The initiator reconnected on the other hand when reconnected within reconnection time-out time judges whether it is the initiator which logged in before bus reset (Step S32), and when it is an initiator different bus reset before, reconnection of the initiator is refused and it returns to the waiting for reconnection.

[0119] When the same initiator as bus reset before has logged in, it judges whether the command block ORB for printing (ORB containing a printing command) was [be / it] under processing at the bus reset generating time (Step S33 of drawing 19). And when it is not [be / it] under processing, a continuation flag is turned OFF (Step S36), and a state transition is carried out to an idle state (Step S37).

[0120] On the other hand, when the command block ORB for printing is under processing, it judges whether bus reset occurred in the light of the status (period after carrying out the light of the status until ACK comes on the contrary) (Step S34). and the information on ACK memorized at Step S24 of drawing 18 when bus reset occurred in the light of the status -- being based -- ACK -- it judges whether it is complete (Step S35)

[0121] and ACK -- in being complete, a continuation flag is turned OFF (Step S36), and it carries out a state transition to an idle state (Step S37) on the other hand -- ACK -- in not being complete, it judges whether it is ACK missing (Step S38) And if it is not ACK missing, nothing will be carried out (Step S39), but in the case of ACK missing, a continuation flag is turned OFF (Step S40), and as drawing 16 explained, a state transition is changed into a dead state (data transfer improper state) (Step S41).

[0122] When it is judged at Step S34 that generating of bus reset is not among the light of the status, it

judges whether at least 1 byte of transfer (printing) data of ORB under processing were transmitted to the latter printer engine (Step S42). And when 1 byte is not transmitted, a continuation flag is turned OFF (Step S43), and a state transition is carried out to an idle state (Step S49).

[0123] On the other hand, when at least 1 byte is transmitted to the printer engine, the contents (data size, the page table existence flag P, command block) of ORB and the size of the data transmitted by the bus reset generating point in time are memorized (Step S44). It is at the byte count of the data which it is the size of this data at the bus reset generating time, and were already transmitted to the latter printer engine, and the bus reset generating time, and the data transfer on the bus of IEEE1394 is already completed, and it is equivalent to the sum total of the byte count of the data which are due to be transmitted to a latter printer engine under a transfer or after this. That is, for example, it is equivalent to the sum total of the byte count of the data of a printing schedule by the byte count of the data already printed by the printer, and the printer under the present printing or after this.

[0124] Next, it judges whether a page table exists (Step S45), and in not existing, it memorizes the contents of the data scripter of ORB (Step S46). That is, when a page table does not exist, the address and the data length of transfer data of direct addressing are memorized (refer to drawing 6 (C)). [of a case]

[0125] On the other hand, when a page table exists, as drawing 14 explained, the contents (the address, data length) of the head segment of a page table, the contents (the address, data length) of the segment which it is at the bus reset generating time, and was under processing and which it was, and a segment number are memorized (Step S47). And the continuation flag explained by drawing 13 is turned ON (Step S48), and a state transition is carried out to an idle state (Step S49).

[0126] Thus, with this operation form, when it is judged that bus reset occurred during the data transfer, (Steps S33 and S34 of drawing 19 , Step S42 of drawing 20), and a continuation flag are turned on [them] (Step S48).

[0127] Drawing 21 and drawing 22 are flow charts which usually show the detailed example of processing at the time.

[0128] First, it judges whether the lead of ORB was directed from the initiator (Step S51), and (was the ring of the doorbell register carried out or not?) when not directed, it remains in an idle state (Step S50). On the other hand, when directed, ORB which the initiator created is led from an initiator (Step S52). And based on the page table existence flag P which ORB contains, it judges whether a page table exists (Step S53). And when a page table exists, it leads eight segments of segments of a page table at a time, for example (Step S54).

[0129] Next, based on the operation code in the command block of ORB, as drawing 15 explained, it judges whether it is the command block ORB for printing of led ORB (Step S55). And it judges whether they are the first eight segments (eight segments containing a head segment) of a page table (Step S56), and when eight segments led at Step S54 when it was the command block ORB for printing are the first eight segments, they shift to the command address comparison processing shown in drawing 22 (Step S57).

[0130] Read/write of data is performed, when it is judged that they are not the first eight segments at Step S56 when it is judged at Step S55 that it is not the command block ORB for printing, and when command address comparison processing of Step S57 is completed (Step S58). And it repeats until it carries out read/write of the data for one segment, and the data for eight segments (Steps S59 and S60).

[0131] Next, when it judges whether read/write of all the segments of a page table was carried out (Step S61) and read/write of all the segments has not been carried out, the eight next segments of a page table are led (Step S54). On the other hand, when all the segments of a page table are led, the light of the status is carried out to an initiator (Step S62). And it judges whether all ORB for printing of printed matter was led (Step S63), when there is following ORB, it returns to Step S52, and when there is no following ORB, a state transition is carried out to an idle state (Step S50).

[0132] When it is judged that a page table does not exist at Step S53, it judges whether it is the command block ORB for printing of led ORB (Step S64). And when it is the command block ORB for printing, it shifts to the command address comparison processing shown in drawing 22 (Step S65).

[0133] When it is judged on the other hand that it is not the command block ORB for printing, and when command address comparison processing is completed, it repeats until it carries out read/write of the data (Step S66) and carries out read/write of all the data (Step S67). And when read/write of all the data is carried out, it shifts to Step S62 and the light of the status is carried out to an initiator.

[0134] In command address comparison processing of drawing 22, a continuation flag judges first whether it is ON (Step S70). This continuation flag is a flag turned ON in Step S48 of drawing 20. And when a continuation flag is OFF, it shifts to Step S76, and the start address (address of the head segment of a page table) of transfer data is memorized, and command address comparison processing is ended.

[0135] When a continuation flag is ON, as drawing 11 and drawing 12 explained, it judges whether the content of led ORB is the same as the content of ORB before bus reset (Step S71). In this case, the content of ORB before the bus reset used as the candidate for comparison is memorized in Step S44 of drawing 20. Moreover, with this operation gestalt, content comparison (Step S71) of ORB is performed in advance of address comparison (Step S72).

[0136] When the content of ORB is the same as that of bus reset before, it judges whether the start address of transfer data is the same as that of bus reset before (Step S72). And in being the same, it returns a setup of data transfer to the state before bus reset generating (Step S73). That is, a setup of data transfer is returned to the state before bus reset so that data transfer can be resumed based on the content, a segment number, etc. of the transmitted data size of the bus reset generating time memorized at Step S44 of drawing 20, and the segment memorized at Step S47 from the position (continuation at the bus reset generating time) shown in D5 of drawing 11. And a continuation flag is returned at OFF (Step S74). In this case, it is made not to clear the data on the data buffer of a target so that the data which had already completed the transfer before bus reset may not disappear as drawing 17 explained.

[0137] In addition, the start address of transfer data is not memorized like Step S76 after Step S74 because the start address memorized before bus reset generating can be used as it is when resuming data transfer from the continuation at the bus reset generating time.

[0138] When it is judged at Step S71 that the content of ORB is not the same as that of bus reset before, or when it is judged at Step S72 that a start address is not the same as that of bus reset before, while not performing the restart procedure of data transfer but returning a continuation flag at OFF, ORB which memorized namely, (Steps S75 and S76) led the start address of transfer data in this case will be processed as entire new ORB.

[0139] In addition, in the case of Step S75, since led ORB will be processed from the beginning, unlike Step S74, the data of the data buffer of a target are cleared.

[0140] 6. Explain the example of electronic equipment, next the electronic equipment containing the data transfer control unit of this operation gestalt.

[0141] For example, the internal-block view of the printer which is one of the electronic equipment is shown in drawing 23 (A), and the external view is shown in drawing 24 (A). CPU (microcomputer) 510 performs system-wide control etc. A control unit 511 is for a user operating a printer. A control program, a font, etc. are stored in ROM516, and RAM518 functions on it as a work field of CPU510. A display panel 519 is for telling a user about the operating state of a printer.

[0142] The printing data sent from partner nodes, such as a personal computer, are directly sent to the printing processing section (printer engine) 512 through a bus 504 through the PHY device 502 and the data transfer control unit 500. And given processing is performed in the printing processing section 512, and printing data are printed by paper by the printing section (equipment for outputting data) 514 which consists of a print header etc., and are outputted.

[0143] The internal-block view of the scanner which is one of the electronic equipment is shown in drawing 23 (B), and the external view is shown in drawing 24 (B). CPU520 performs system-wide control etc. A control unit 521 is for a user operating a scanner. A control program etc. is stored in ROM526 and RAM528 functions as a work field of CPU520.

[0144] The picture of a manuscript is read by the picture reading section (equipment for incorporating data) 522 which consists of the light source, an optical/electrical converter, etc., and the data of the read picture are processed by the image-processing section (scanner engine) 524. And the image data after

processing is directly sent to the data transfer control unit 500 through a bus 505. The data transfer control unit 500 generates a packet by adding a header etc. to this image data, and transmits it to partner nodes, such as a personal computer, through the PHY device 502.

[0145] The internal-block view of the CD-RW drive which is one of the electronic equipment is shown in drawing 23 (C), and the external view is shown in drawing 24 (C). CPU530 performs system-wide control etc. A control unit 531 is for a user operating CD-RW. A control program etc. is stored in ROM536 and RAM538 functions as a work field of CPU530.

[0146] The data which consist of laser, a motor, optical system, etc. and which read and were read in CD-RW532 by the & write-in section (equipment for memorizing the equipment or data for incorporating data) 533 are inputted into the signal-processing section 534, and given signal processing, such as error correction processing, is performed. And the data with which signal processing was performed are directly sent to the data transfer control unit 500 through a bus 506. The data transfer control unit 500 generates a packet by adding a header etc. to this data, and transmits it to partner nodes, such as a personal computer, through the PHY device 502.

[0147] On the other hand, the data sent from the partner node are directly sent to the signal-processing section 534 through a bus 506 through the PHY device 502 and the data transfer control unit 500. And given signal processing is performed to this data by the signal-processing section 534, it reads, and CD-RW532 memorizes by the & write-in section 533.

[0148] In addition, you may make it prepare independently CPU for the data transfer control with the data transfer control unit 500 other than CPUs 510, 520, and 530 in drawing 23 (A), (B), and (C).

[0149] Moreover, although RAM501 (equivalent to a data buffer) is formed in the exterior of the data transfer control unit 500 in drawing 23 (A), (B), and (C), you may make RAM501 build in the data transfer control unit 500.

[0150] When using the data transfer control unit of this operation gestalt for electronic equipment, and new electronic equipment is connected to a bus and bus reset occurs, generating of the fault which considers bus reset as a cause is prevented. Thereby, the malfunction of electronic equipment can be prevented.

[0151] Moreover, if the data transfer control unit of this operation gestalt is used for electronic equipment, high-speed data transfer will become possible. Therefore, when a user directs print-out with a personal computer etc., printing comes to be completed in few time lag. Moreover, it can read in few time lag after directions of the picture incorporation to a scanner, and a user can see a picture now. Moreover, reading of the data from CD-RW and the data to CD-RW can be written in now at high speed.

[0152] Moreover, it becomes possible to mitigate the processing burden of the firmware which operates on CPU, and to use the bus of cheap CPU or a low speed by using the data transfer control unit of this operation gestalt for electronic equipment. Furthermore, since low-cost-izing of a data transfer control unit and small-scale-ization can be attained, low-cost-izing of electronic equipment and small-scale-ization can also be attained.

[0153] In addition, as electronic equipment which can apply the data transfer control unit of this operation gestalt, various things, such as various optical disk drives (CD-ROM, DVD), a Magnetic-Optical disk drive (MO), a hard disk drive, TV and VTR, a video camera, audio equipment, telephone, a projector, a personal computer, an electronic notebook, and a word processor, can be considered besides the above.

[0154] In addition, this invention is not limited to this operation gestalt, but deformation implementation various by within the limits of the summary of this invention is possible for it.

[0155] For example, although the composition of the data transfer control unit of this invention has especially the desirable composition shown in drawing 7, it is not limited to this.

[0156] Moreover, although the judgment technique of whether bus reset occurred during the data transfer, the comparison technique of a command, and the technique of resumption of data transfer also have especially the desirable technique explained with this operation gestalt, it is not limited to this.

[0157] Moreover, although this invention is useful to especially the bus reset in IEEE1394, it is

applicable if it is the reset which clears the topology information on a node at least besides this.
[0158] Moreover, although especially the thing of this invention applied to the data transfer in IEEE1394 specification is desirable, it is not limited to this. For example, this invention is applicable also to the data transfer in the specification based on the same thought as IEEE1394, or the specification into which IEEE1394 was developed.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is drawing showing the layer structure of IEEE1394.

[Drawing 2] It is drawing for explaining SBP-2.

[Drawing 3] It is drawing for explaining the outline of data transfer processing of SBP-2.

[Drawing 4] It is drawing for explaining command processing in the case of transmitting data to a target from an initiator.

[Drawing 5] It is drawing for explaining command processing in the case of transmitting data to an initiator from a target.

[Drawing 6] Drawing 6 (A), (B), and (C) are drawings for explaining a page table.

[Drawing 7] It is drawing showing the example of composition of the data transfer control unit of this operation gestalt.

[Drawing 8] It is the flow chart which shows the outline of processing by the side of a target (firmware).

[Drawing 9] It is the flow chart which shows the outline of processing by the side of an initiator (device driver).

[Drawing 10] Drawing 10 (A), (B), and (C) are drawings for explaining the problem of double printing.

[Drawing 11] It is drawing for explaining the technique which bus reset generates during the data transfer, and continues data transfer and is resumed before and after bus reset when the content of ORB is the same.

[Drawing 12] It is drawing for explaining content comparison of ORB.

[Drawing 13] It is drawing for explaining a continuation flag.

[Drawing 14] It is drawing for explaining the information for specifying the address which resumes data transfer.

[Drawing 15] It is drawing for explaining the technique of transferring command comparison processing until the first command block ORB containing a printing command comes.

[Drawing 16] When bus reset occurs in the light of the status and it becomes ACK missing, it is drawing for explaining the technique of shifting to a dead state.

[Drawing 17] It is drawing for explaining the technique held without canceling the data which have not yet been transmitted to an initiator at the time of bus reset.

[Drawing 18] It is the flow chart which shows the detailed example of processing of this operation form at the time of bus reset generating (at the time of reconnection).

[Drawing 19] It is the flow chart which shows the detailed example of processing of this operation gestalt at the time of bus reset generating (at the time of reconnection).

[Drawing 20] It is the flow chart which shows the detailed example of processing of this operation gestalt at the time of bus reset generating (at the time of reconnection).

[Drawing 21] Usually, it is the flow chart which shows the detailed example of processing of this operation gestalt at the time.

[Drawing 22] Usually, it is the flow chart which shows the detailed example of processing of this

operation gestalt at the time.

[Drawing 23] Drawing 23 (A), (B), and (C) are the examples of the internal-block view of various electronic equipment.

[Drawing 24] Drawing 24 (A), (B), and (C) are the examples of the external view of various electronic equipment.

[Description of Notations]

10 Data Transfer Control Unit

12 PHY Device

14 Link Device

16 CPU

18 Data Buffer

20 Firmware (F/W)

30 Communication Section (COM)

40 Management Section (MNG)

50 Print Task Section (PRT)

60 Fetch Section (FCH)

62 Judgment Section

64 Command Storage Section

66 Command Comparator

68 Address Storage Section

70 Address Comparator

72 Data Transfer Resumption Section

100 Personal Computer

102 Device Driver

104 Data Buffer

110 Information-Storage Medium

[Translation done.]